MSS54

Modulbeschreibung Momentenmanagement

CHANGE DOCUMENTATION FROM R300

Version	date	comment
S300	06/01/2004	Transfer from MSS60 project
S310	10/2/2004	Changed the overview image for EVT filling control and added the control edges module
S320	11/16/2004	Document roughly revised (due to lack of time due to software requirements). Ignition angle degrees temporarily set to 100% in the SW
S330	1.12.2004	Verbrauchsmoment bei Generatorbetrieb des KSG in die Momentenstruktur included
S330	1.12.2004	Calculation wi moved from operating mode manager to torque manager
S350	February 13, 2005	Document completely revised again
S360	10.3.2005	B_EVT removed because only EVT motors are served
S370	07/04/2005	Complete torque conversion of the desired torque to the negative range
S370	07/06/2005	Complete conversion of the LS/Dashpot filter and removal of the SA/WE filter
S370	08/30/2005 rm	Calculation of ignition angle engagement / ignition hook
S370	9/11/2005	Actual torque calculation changed
S370	9/18/2005	Torque calculation from HFM signal (according to spec. by F. Mayer)
S380	5.11.2005	Dynamic filter expanded by two areas
S380	11/5/2005	Calculation of idle controller adaptation md_llra is now subtracted instead of added
S380	11/16/2005	Invoicing of md_e_verbraucher completely changed

1. OVERVIEW TORQUE MANAGER

With the conversion of all engine torque interventions (ASC/DSC, EGS, IHKA) to a standardized torque interface, central coordination of all torque requirements for the filling and ignition path became necessary. The torque manager should take over this task.

The following list is intended to give a brief overview of the individual modules of the torque manager or modules that are very closely linked to it. For reasons of clarity, the description is reduced to a core sentence and only the most important variables for the sequence of the torque calculation are limited as input and output variables.

Module: pedal value acquisition
Determination of the relative driver's desired torque
Input variables: pwg1, pwg2, n, S_FDYN
Output variables: pwg_soll

Module: Minimum Moment
Determination of the minimum indicated engine torque
Input variables: n, state_motor, md_llra, md_reib_filter, md_e_consumer
Output variables: md_e_schlepp, md_e_schlepp_hyp

Module: Maximum Moment
Determination of the maximum indicated engine torque
Input variables: n, md_e_schlepp, rf_pt_korr, md_fw_rel
Output variables: md_e_max

Module: consumer
Determination of consumer moments
Eingangsgrößen: can_kkos_lm, md_ksg, S_KO
Output variables: md_e_ Verbraucher

Module: Reibmoment
Determination of the friction torque of the engine
Input variables: n, tmot, toel
Output variables: md_reib_filter

Module: Calculation of the driver 's desired torque
Determination of the absolute driver's desired torque including FGR
Input variables: md_fw_rel, md_ind_fgr, b_fgr_aktiv, md_e_schlepp_hyp Output variables: md_e_fw, d_md_ Wunsch_rel

Module: Dynamic filter
Filtering driver's desired torque
Input sizes: md_e_fw, sa_we_st, dyn_st, gang, md_e_schlepp
Output variables: md_fw_filter, md_sawe_filter, md_ind_wish

Module: Idle speed control intervention
Consideration of the I component of the idling control
Input variables: md_ind_wish, md_llri
Output variables: md_ind_wish_filter (actually the name is not correct)

Module: Vmax limitation gear-
dependent Vmax limitation
Input variables: v_drive, d_v, gear
Output variables: md_ind_vmax, vmax_st

Module: Torque limitation

torque limitation
Input variables: gang, B_positive connection, d_n, md_ind_schlepp, md_eta_zw_ve
Output variables: md_max_begr, md_begr_st

Module: Momentenbegrenzung

Coordination of the torque limitations
Input variables: md_ind_vmax, vmax_st, md_max_begr, md_sk_begr, sk_egas_status, n, d_n_segment, gear

Output variables: md_ind_wish_limit

Module: Reserve moments

Creation of a torque reserve for cat heating
Input variables: kath_status, n, wi, tmot, t_ml
Output variables: md_res

Module: DSC intervention

Torque interventions of the DSC system in the filling path
Input variables: asc_st, md_ind_asc, md_ind_msr
Output variables: md_ind_asc_abs, md_ind_msr_abs

Module: implementation in filling by control edges

Implementation of the torque specification in control edges, basic throttle valve angle
Input variables: wi, n, rf_pt_korr
Output variables: ao_aw, as_aw, eo_aw, es_aw, ml_soll_bas, wdk_soll_evt, ti_ende_evt

Module: Momenteneingriffe Zündwinkel

Coordination of the torque interventions in the ZW path
Input variables: md_ind_wish_begr, md_llr_tz, md_ind_asc_abs, md_ind_msr_abs
Output variables: md_tz_red

Module: Calculation of ZW intervention

Calculation of an absolute ignition angle based on the torque specification and the actual torque of the engine

Input variables: md_tz_red, md_ind_opt_korr, md_eta_zw_min, n, wi
Output variables: tz_md[x]

Module: Calculation of base sizes

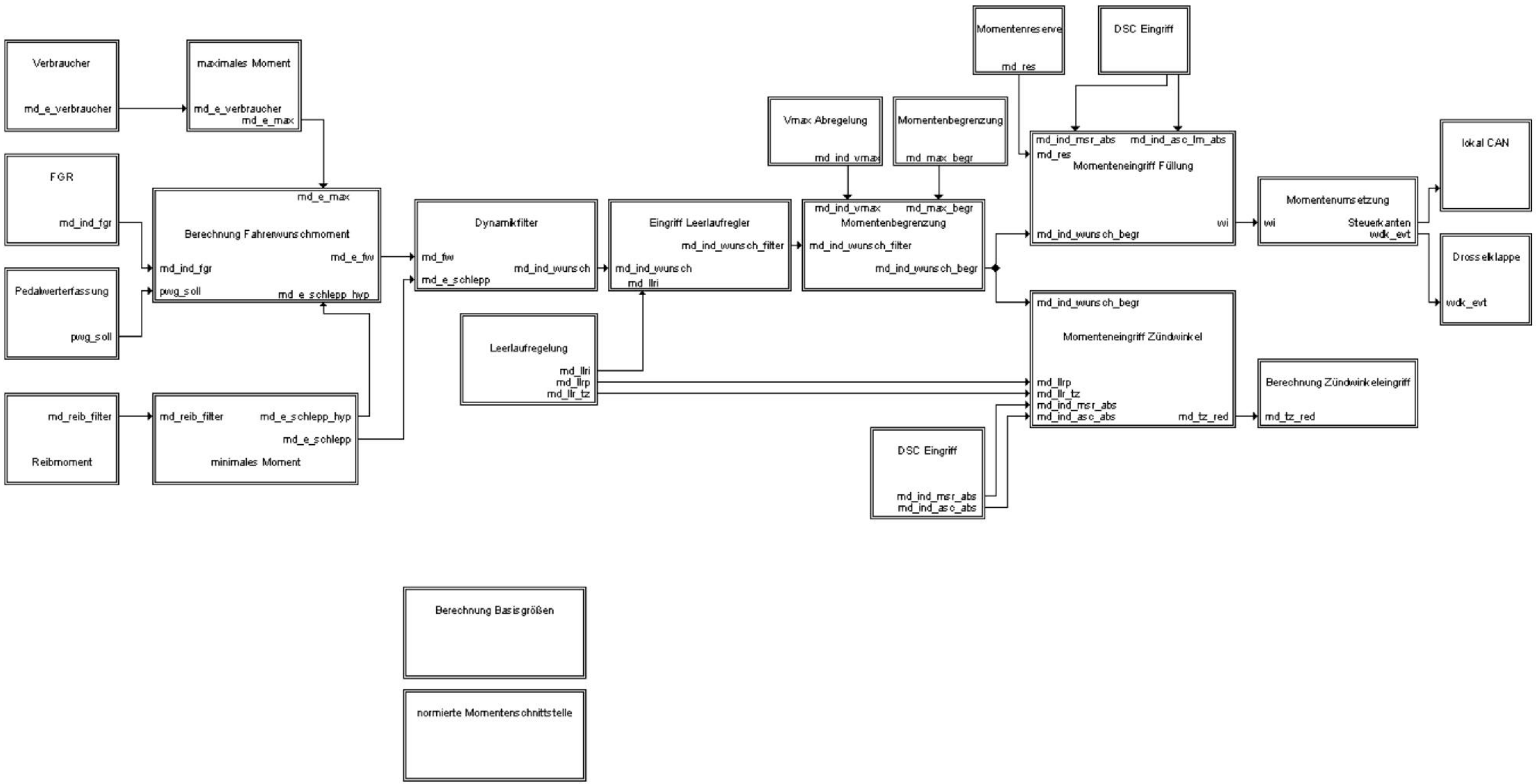
Module for calculating the different actual torques and ZW efficiencies, as well as all
Auxiliary variables for torque calculation and coordination

Module: normalized torque interface

Implementation of the standardized torque interface and the air conditioning compressor activation according to CAN specifications 11H

Machine Translated by Google

Image: Total overview of the torque manager (mm.gif)



2. CALCULATION OF BASE VARIABLES

A main component of the torque manager is the determination of the actual torques before and after torque interventions, the optimal ignition angle, and the calculation of the efficiency of lambda and ignition angle corrections. The calculation of the individual variables is summarized in the "Calculation of basic variables" module.

The indicated actual torque corresponds to the torque applied to the clutch and the friction. The actual torque is first calculated in order to switch consumers (or disturbance variables) on or off in a torque-neutral manner. It is a smoothed delayed wi target specification based on data that has been driven out.

2.1. TORQUE CALCULATION

Corrected maximum actual wi "md_wi_opt_korr"

The corrected maximum indexed actual wi "md_wi_opt_korr" takes into account the lambda influence on the generated engine torque. It corresponds to the actual torque "md_wi_ind_opt_th", corrected by the current lambda efficiency "md_eta_lambda".

md_wi_opt_korr = md_wi_ind_opt_th * md_eta_lambda

Actual actual torque before torque interventions "md_ind_ve"

The torque "md_ind_ve" represents the actual torque generated by the engine, which it would deliver without the torque manager's ignition angle intervention. However, the torque reductions caused by the ignition angle interventions of other modules such as knock control, cat heating, etc. are taken into account. This takes place in the form of an ignition angle efficiency "md_eta_zw_ve", the calculation of which is also described in this chapter.

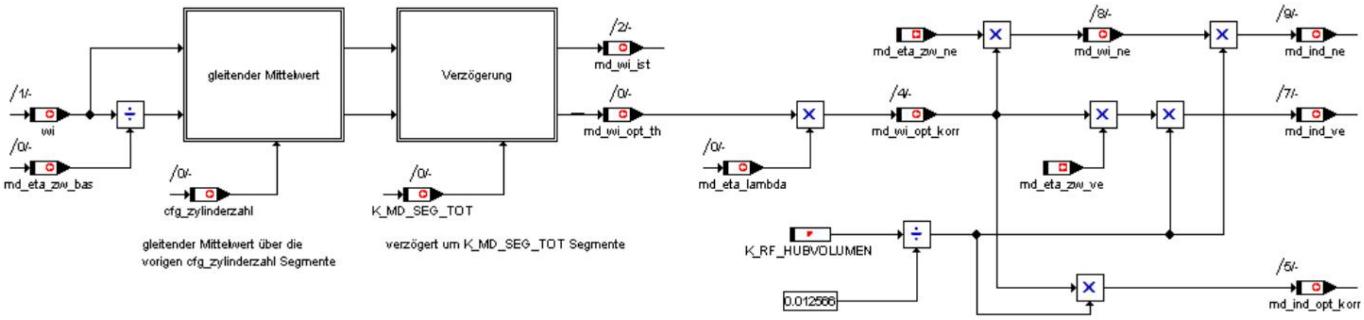
md_ind_ve = (md_wi_opt_korr * md_eta_zw_ve) * K_RF_HUBVOLUMEN / 0.012566

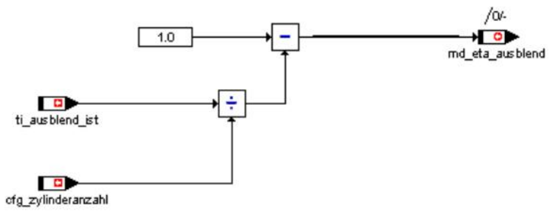
Actual actual torque after torque interventions "md_ind_ne"

The torque "md_ind_ne" represents the actual torque generated by the engine, taking into account all torque interventions. For this purpose, an ignition angle efficiency "md_eta_zw_ne" is taken into account, which also includes the ignition interventions of the torque manager.

md_ind_ne = (md_wi_opt_korr * md_eta_zw_ne) * K_RF_HUBVOLUMEN / 0.012566

Image: Calculation of actual torques (md_ist.gif)





Description of the actual torque calculation

The setpoint "wi" is delayed by the applicable segment counter K_MD_SEG_TOT, which corresponds to the segment dead time. This size is also smoothed with the wi calculated for the last "cfg_cylinder number" segments. The resulting variable "md_wi_ist" corresponds to the stationary "wi" that was driven out on the test bench, which is knock-limited in some points or includes an advance ignition angle (in the idling range). The theoretical best value of the ignition hook "md_wi_opt_th" is determined with the efficiency "md_eta_zw_bas". The lambda influence is taken into account in the "md_wi_opt_korr" variable. Then the instantaneous "md_wi_ne" is calculated with the efficiency "md_eta_zw_ne" after all ignition angle interventions and with lambda influence. Since S370, Md_eta_ausblend is no longer used to calculate "md_ind_ne" (md_wi_ne) because incorrect values would be calculated in the case of cylinder deactivation.

md_eta_zw_bas: Efficiency pre-control ignition angle (applied in the map) to theoretical
Best ignition angle without knock limitation and torque advance md_zw_opt_korr,
since the theoretical best ignition angle is temperature-dependent and
Is changed depending on the lambda, the pre-control ignition angle should be changed
with the physically similar facial expressions

md_eta_zw_ne: Efficiency of current ignition angle to theoretical best ignition angle without knock limitation

md_eta_lambda: Efficiency influence lambda (full load, component protection, ...)

delayed and smoothed wi:

wi delayed by K_MD_SEG_TOT segments (180 degrees for 4 cylinders) and moving average over the previous cfg_cyl number segments.

Example: cfg_cylinder number = 4, K_MD_SEG_TOT = 5

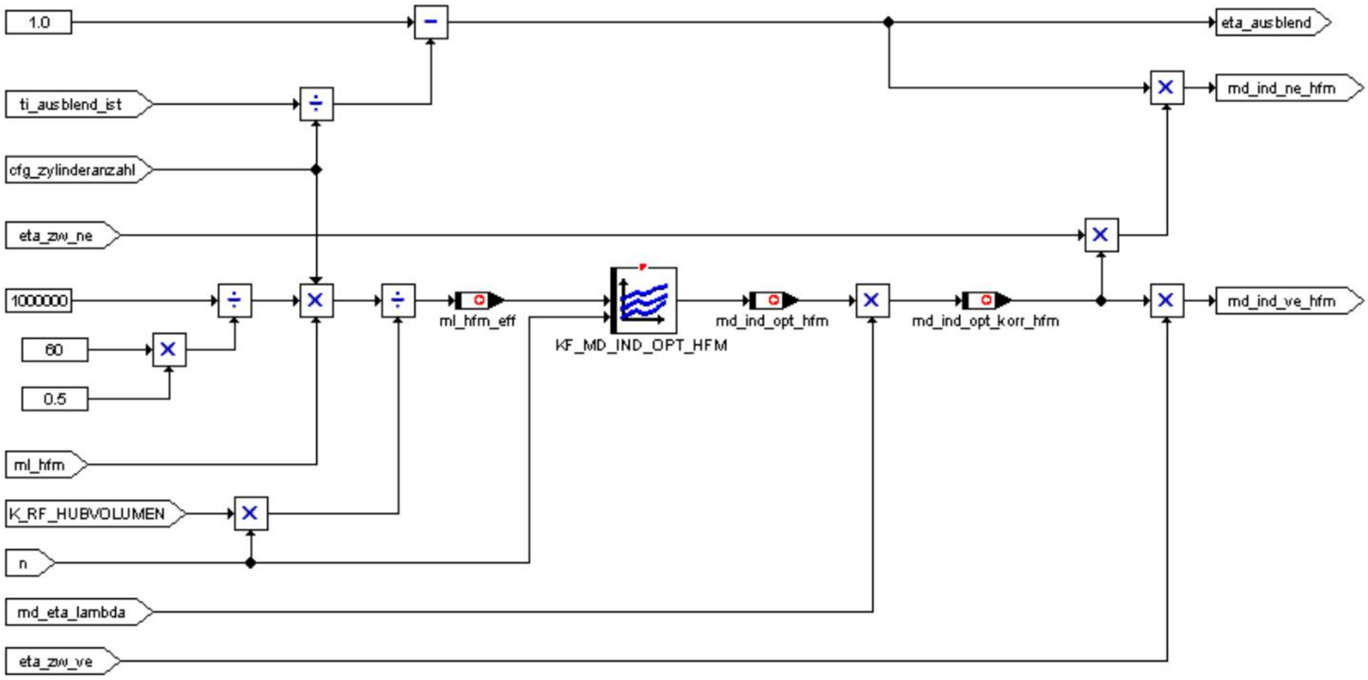
Target specifications: wi(1) wi(2) wi(3) wi(4) wi(5) wi(6) wi(7) wi(8) wi(9) wi(10)

Actual torque in the 10th segment: md_wi_ist(10) = (wi3+wi4+wi5+wi6)/4

Stability calculation from HFM signal

physical background
In order to be able to make defined torque interventions, the currently set engine torque must be available in the function structure. Under the boundary conditions of optimal ignition angle and Lambda = 1, the air mass drawn in by the engine is directly proportional to the engine torque.
Taking efficiency interventions into account, the engine torque can thus be stored in a map of engine speed and air mass and made available to the functional structure.

Image: Calculation of actual torques (md_ist_hfm.gif)



Optimum indexed actual torque from hfm

The optimal indicated actual torque md_ind_opt_hfm is the torque that the engine generates at the operating point with optimal ignition timing and lambda=1. The actual torque was determined on the test bench as a function of the speed and the air mass measured by the HFM, per work cycle and cylinder in relation to the cylinder displacement, and stored in the map KF_MD_IND_OPT_HFM.

(1) md_ind_opt_hfm = KF_MD_IND_OPT_HFM (n, ml_hfm_eff)

Corrected optimal indexed actual torque from hfm

The corrected optimal indexed actual torque md_ind_opt_korr_hfm takes into account the lambda influence on the generated engine torque. It corresponds to the actual torque md_ind_opt_hfm, corrected by the current lambda efficiency md_eta_lambda.

(2) md_ind_opt_korr_hfm = md_ind_opt_hfm * md_eta_lambda

actual actual torque before torque interventions from hfm

The torque md_ind_ve_hfm represents the actual torque actually generated by the engine, which it would deliver without the torque manager intervention in the ignition angle. However, the torque reductions caused by the ignition angle interventions of other engine modules such as knock control, cat heating, etc. are taken into account. This takes place in the form of an ignition angle efficiency eta_zw_ve (see Chapter 1.1.2.4 "Calculation of ignition angle interventions").

(3) md_ind_ve_hfm = md_ind_opt_korr_hfm * md_eta_zw_ve

actual actual torque after torque interventions from hfm

The torque md_ind_ne_hfm represents the actual torque generated by the engine, taking into account all torque interventions. For this purpose, an ignition angle efficiency eta_zw_ne is taken into account, which also includes the ignition interventions of the torque manager (chap.
"Calculation of ignition angle intervention"). Furthermore, injection suppressions of individual or all cylinders are also included in the calculation in the form of a suppression efficiency eta_suppression.

(4) md_ind_ne_hfm = md_ind_opt_korr_hfm * md_eta_zw_ne * md_eta_ausblend

Machine Translated by Google

2.1.1. CALCULATION OF LAMBDA EFFICIENCY

In addition to the ignition angle, the lambda ratio also has an influence on the indicated engine torque. All torque maps have been determined for a lambda of 1.0. In real engine operation, the lambda ratio that is actually present must be determined and the corresponding actual and desired torques must be corrected with a correction factor.

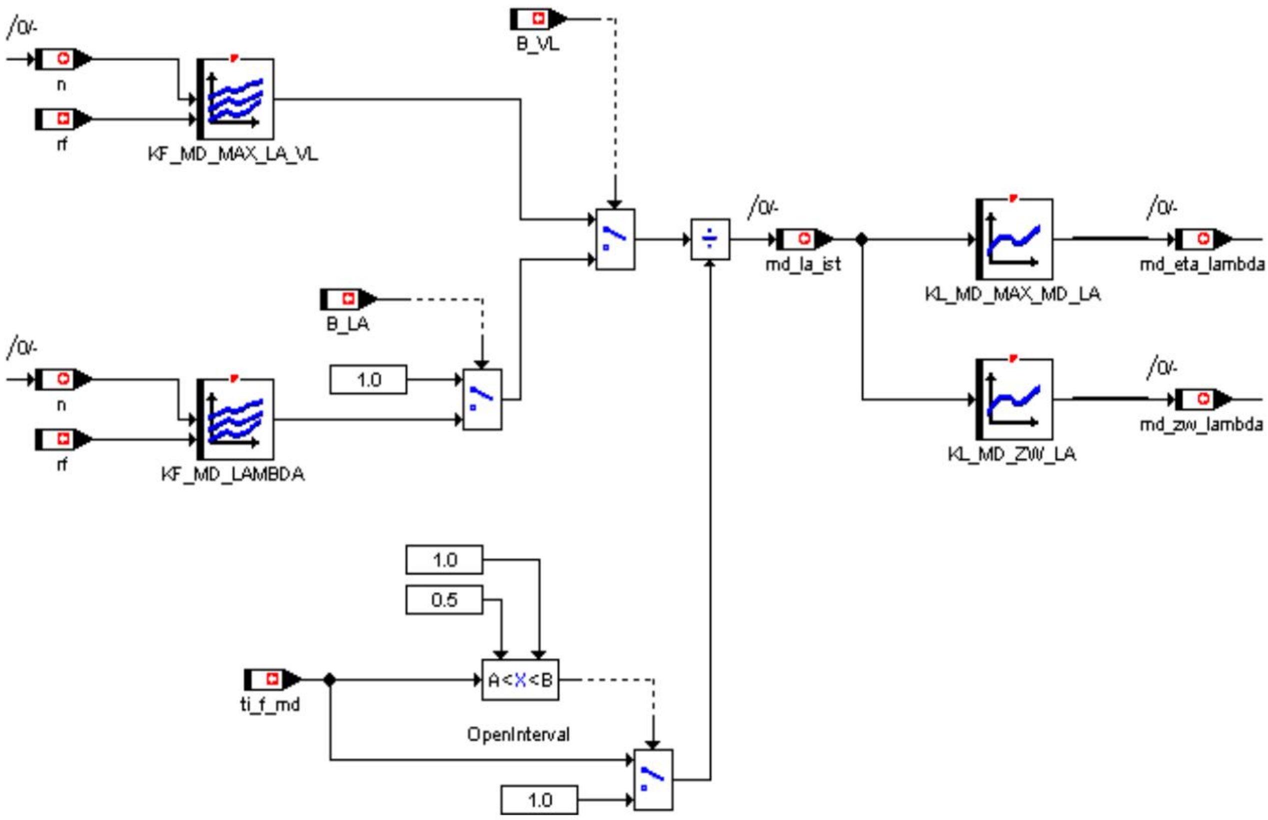
In the lambda-controlled range, lambda is always one and therefore the correction factor is also equal to 1.0. In full-load operation, the lambda value is taken from the “KF_MD_MAX_LA_VL” map and converted into a correction factor via the “KL_MD_MAX_MD_LA” characteristic.

In the case of inactive lambda control (eg during warm-up), lambda values unequal to one may also exist due to the pre-control of the mixture, which must be taken into account in the torque path. To do this, the lambda value valid for the operating point must be stored in the "KF_MD_LAMBDA" map.

Mixture leaning during the warm-up phase is taken into account by dividing the lambda value from the characteristic curves by the leaning factor "ti_f_md". Mixture enrichment (ti_f_md > 1) || (ti_f_md < 0.5) are not corrected.

For reasons of completeness, the following graphic also includes the calculation of "md_eta_lambda". This offset ignition angle reflects the influence of the lambda value on the optimum ignition angle.

Image: Calculation of lambda efficiency (lambda effectivity.gif)



3. MOMENTENSCHNITTSTELLE (CAN)

The torque manager currently has three interfaces via CAN to other systems.

3.1. INTERFACE TO CLIMATE CONTROL AND KSG

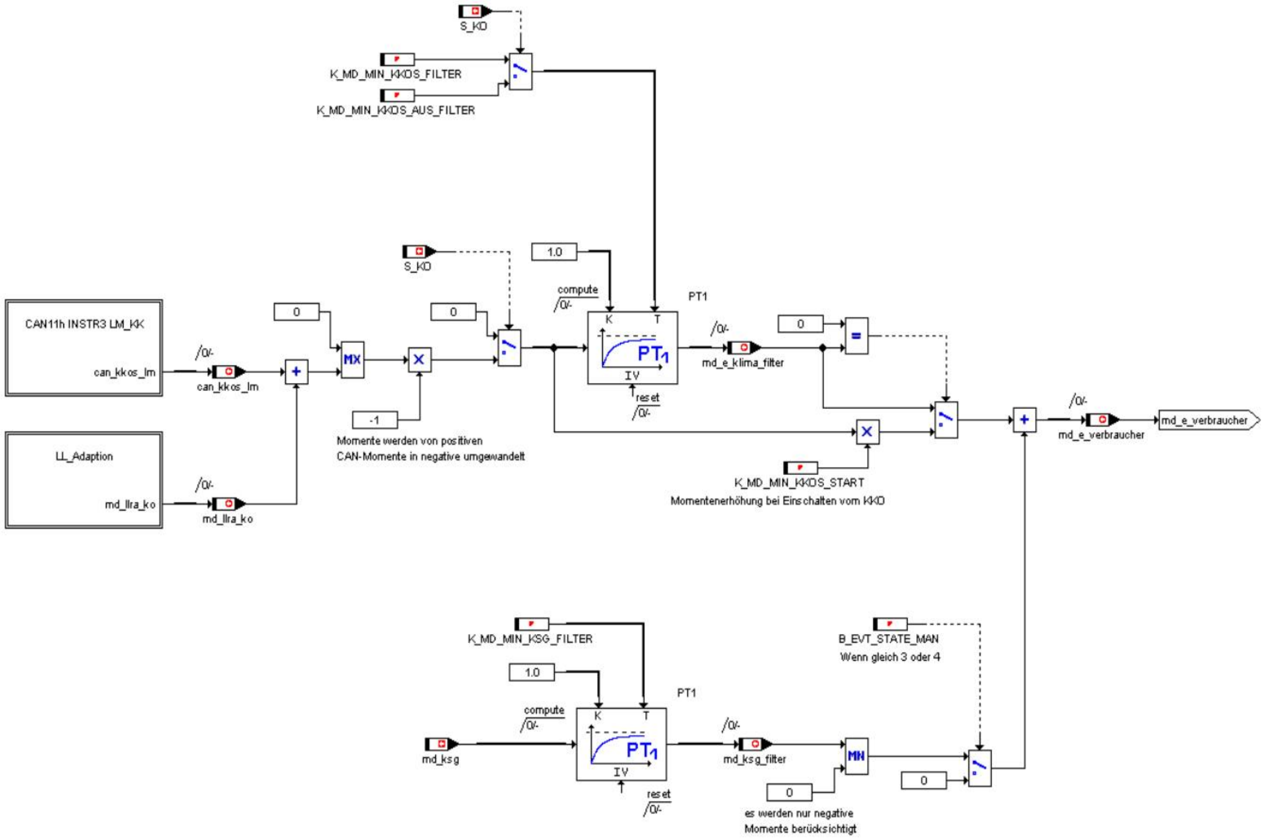
In the "can_kkos_lm" signal, the climate control transmits the current power consumption of the air conditioning compressor. The torque manager must take this torque loss into account in the form of a consumer torque "md_e_ Verbraucher" when calculating the driver's desired torque. Since the torque requirement and the actual torque requirement do not always match exactly, the difference is compensated for by means of a torque adaptation by the idling control.

When a torque request from the air conditioning control is first detected, the requested torque (sum of air conditioning request plus adaptation) is weighted with the factor "K_MD_MIN_KKOS_START", whereby this factor can also be greater than one, which is equivalent to an initial value increase. This initial value is then brought to the requested value via a PT1 filter with the time constant "K_MD_MIN_KKOS_FILTER".

When the air conditioning compressor is switched off, the torque loss is limited to zero with the filter time constant "K_MD_MIN_KKOS_AUS_FILTER".

In addition, the filtered torque from the KSG "md_ksg_filter" in generator mode and with the switching position of "B_EVT_STATE_MAN" (= 3 or 4) is added. "Md_ksg" is transmitted to the engine control via local CAN. The pt1 filtering of "md_ksg" can be influenced with the time constant "K_MD_MIN_KSG_FILTER".

Image: Calculation of torque loss (md_verbraucher.gif)



3.2. INTERFACE TO ASC/DSC - REQUEST TORQUE INTERVENTION

The ASC or DSC control unit is able to influence the indicated engine torque using a standardized torque interface. Three paths are provided as intervention options.

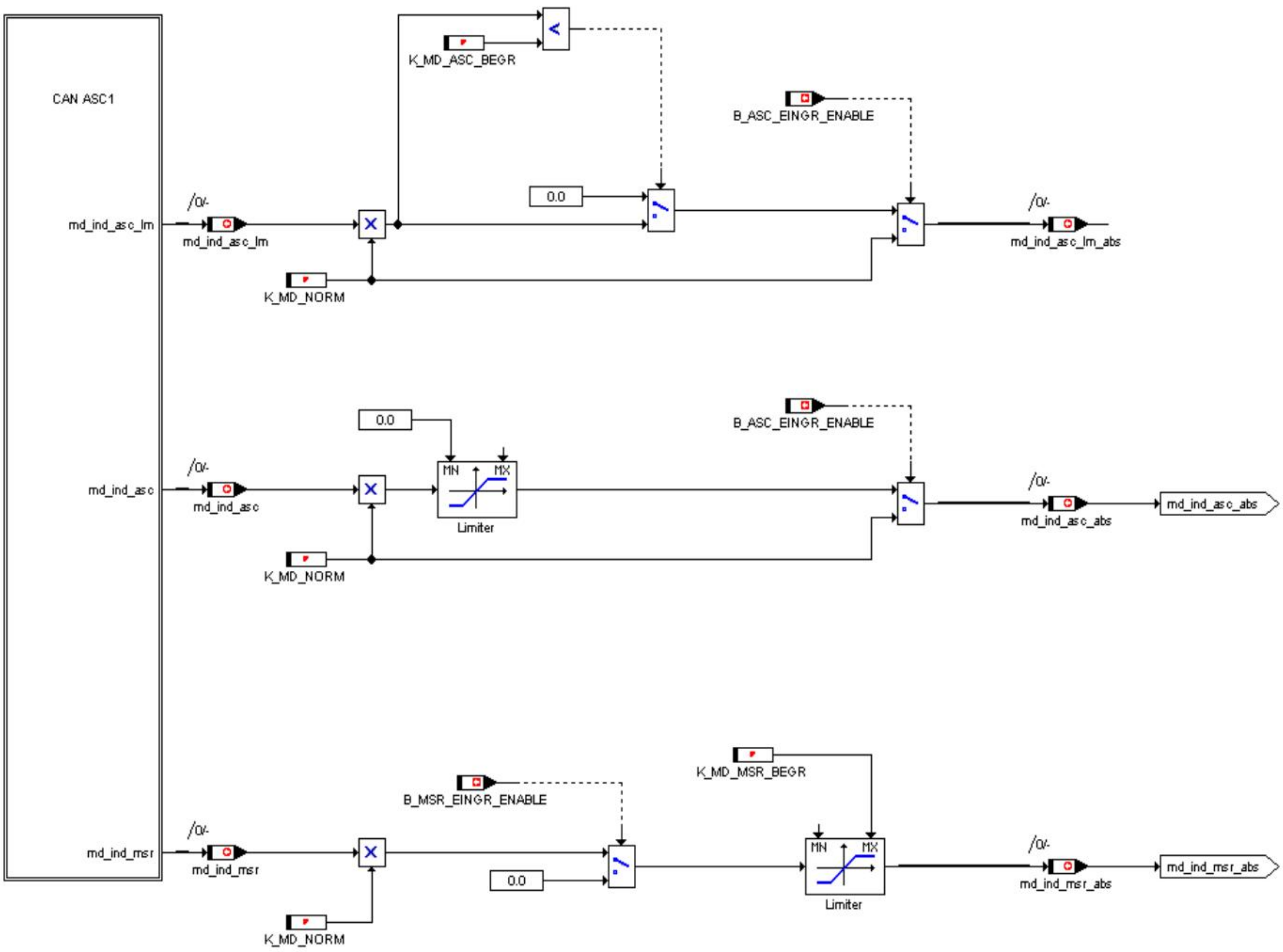
- md_ind_asc_lm: Torque reduction by reducing the filling
- md_ind_asc: - Torque reduction by retarding the ignition angle
md_ind_msr: Torque increase by increasing the filling

All torque requirements are related to an indexed standard torque "K_MD_NORM", the value range is between 0% and 99.6%.

The interventions of the ASC/DSC system can be blocked using the application constant "K_MD_ASC_CONTROL".
Bit 0 = 1 : ASC intervention disabled (corresponds to B_ASC_EINGR_ENABLE = 0)
Bit 1 = 1 : MSR intervention disabled (corresponds to B_MSR_EINGR_ENABLE = 0)

The MSR intervention is limited to the moment "K_MD_MSR_BEGR".

Image: ASC / DSC interface (asc_dsc.gif)



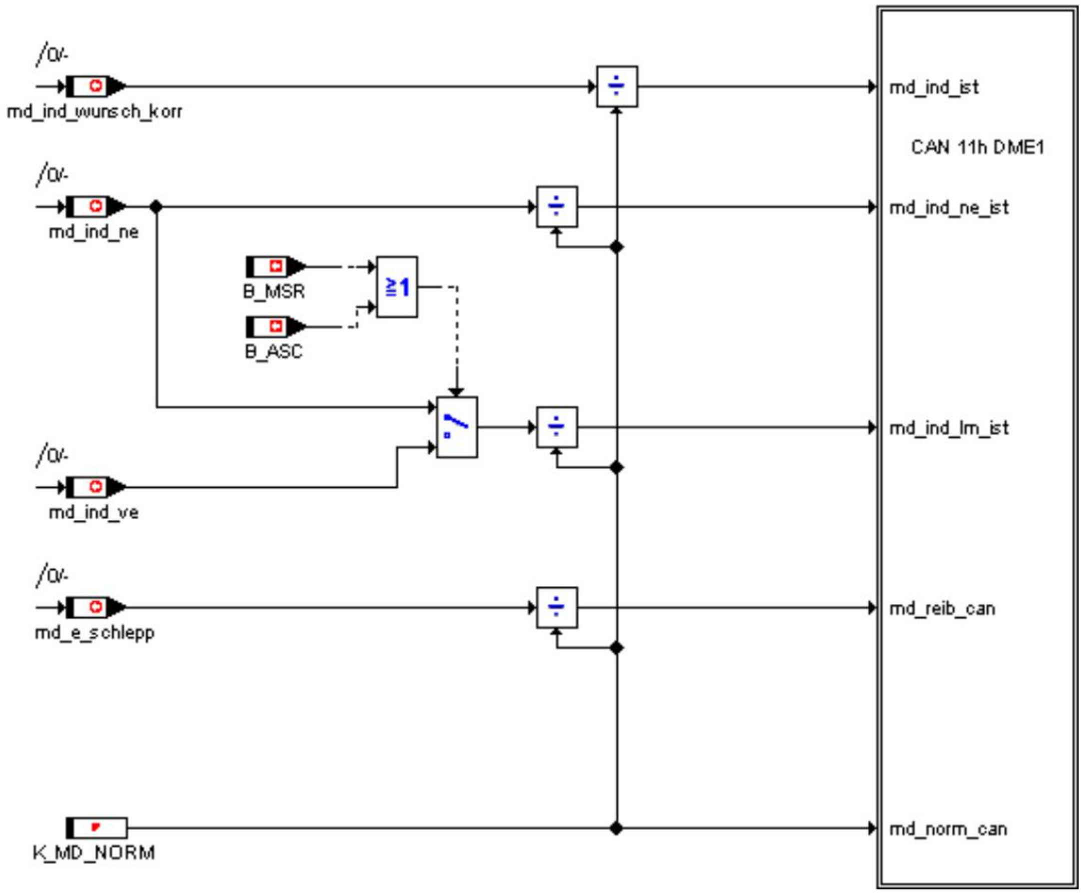
3.3. INTERFACE TO ASC/DSC - FEEDBACK MOMENTARY INTERVENTION

The DME transmits the following torque information back to the DSC:

- md_norm_can :	Reference torque for all torque specifications
- md_reib_can :	Torque loss of the motor including all consumers (alternator, oil pump, air conditioning compressor, ...)
- md_ind_ist :	generated indexed actual torque of the motor without consideration of DSC interventions
- md_ind_ne_ist :	generated indexed actual torque of the motor under consideration of all interventions
- md_ind_lm_ist :	theoretical engine torque, calculated from the measured one Air mass without taking into account the external ignition angle interventions

Since the MSS60 does not differentiate between internal and external ignition angle interventions, the calculation of "md_ind_lm_ist" is not that easy. If no external ZW intervention is active, the moment "md_ind_ne" is used as "md_ind_lm", which also takes into account all internal ZW interventions. If, on the other hand, an external ZW intervention (ASC, MSR) is active, "md_ind_ve" is used, which includes the internal ZW influences from the basic ignition angle, knock control, knock adaptation and dynamic derivative action.

Image: Feedback to DSC (rueckmeldungdsc.gif)



4. FRICTION TORQUE

The friction torque is the torque that is required to turn the engine without firing.

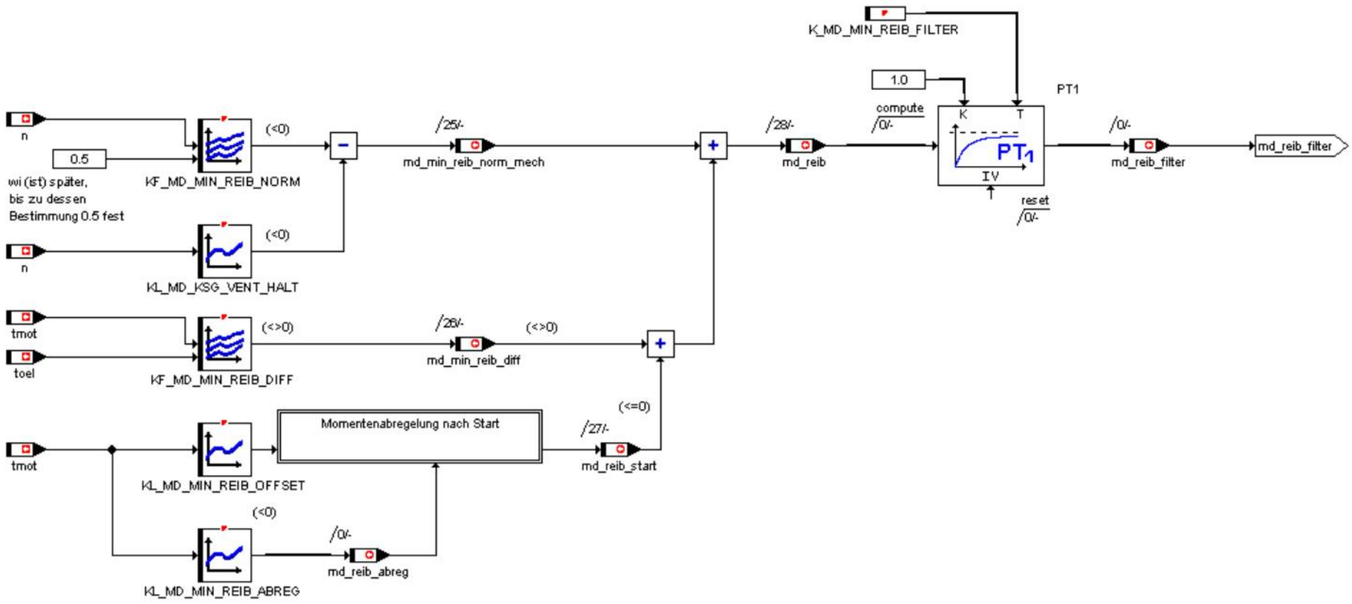
The friction curve is driven out under defined conditions "KF_MD_MIN_REIB_NORM". Deviations from these standard conditions are taken into account in "KF_MD_MIN_REIB_DIFF". With the characteristic curves "KL_MD_MIN_REIB_OFFSET" and "KL_MD_MIN_REIB_ABREG" the additional torque requirement at the start, which is limited, is taken into account.

In the "KF_MD_MIN_REIB_NORM" map, the friction curve is determined at "tmot"=80 ÿC and "toel"=80 ÿC and closed valves. The torque required to lock the valves is stored in the "KL_MD_KSG_VENT_HALT" characteristic. "Md_min_reib_norm_mech" is the drag torque of the engine at defined temperatures without the electrical friction component of the valve train.

Deviations from the standard temperatures are taken into account in the "KF_MD_MIN_REIB_DIFF" map (negative values for colder temperatures, positive values for warmer temperatures).

In order to meet an increased torque requirement of the engine at the start and in the first few seconds afterwards, the drag torque is increased during the start by the offset "md_reib_offset" (KL_MD_MIN_REIB_OFFSET = f (tmot)), which after the end of the restart with the torque ramp "KL_MD_MIN_REIB_ABREG " = f (tmot) is regulated to zero.

Image: Calculation of friction torque (md_reib.gif)



5. DRAG TORQUE

The drag torque is the minimum torque that can be requested by the engine control.
It contains the temperature-dependent basic friction of the engine. Other sizes will be charged therein.

The drag torque on an effective basis is calculated twice. Together, the friction torque "md_reib" is filtered using a PT1 filter and the torque from the idle controller adaptation "md_llra" is subtracted. The differences are as follows:

With "md_e_schlepp_hyp" the moment of the maximum possible gas exchange losses "KL_MD_LW_MIN" is added. The sum is then multiplied by a hyperbolic function so that the drag torque is increased at low speeds (zero effective torque at idle speed). The torque can also be changed manually using the "KL_MD_MIN_FAK_MAN_LLR" characteristic curve in order to achieve an increase in the low speed range.

With "md_e_schlepp" the additional torque "md_min_start" required for the start is added, which depends on the speed and tmot.

In the "KF_MD_MIN_START" map, additional torque values that are required for starting are applied, which are dependent on engine speed and cooling water temperature. Deviating from each other, the following sizes are charged. In the "KL_MD_LW_MIN" characteristic curve, the torque values that can also be achieved with the maximum possible gas exchange losses are stored above the speed. Such heavy losses are created by allowing the engine to compress the cylinder volume and then opening the valves so as not to use any expansion energy.

To calculate "md_e_schlepp_hyp", "md_temp3" is multiplied by a hyperbolic function ("md_min_fak_man_llr") in the low engine speed range, which increases the torque at idle speed to zero effective torque. In the hyperbolic function, the current speed, the target idling speed and the "n_hyp" factor are calculated. For this applies:

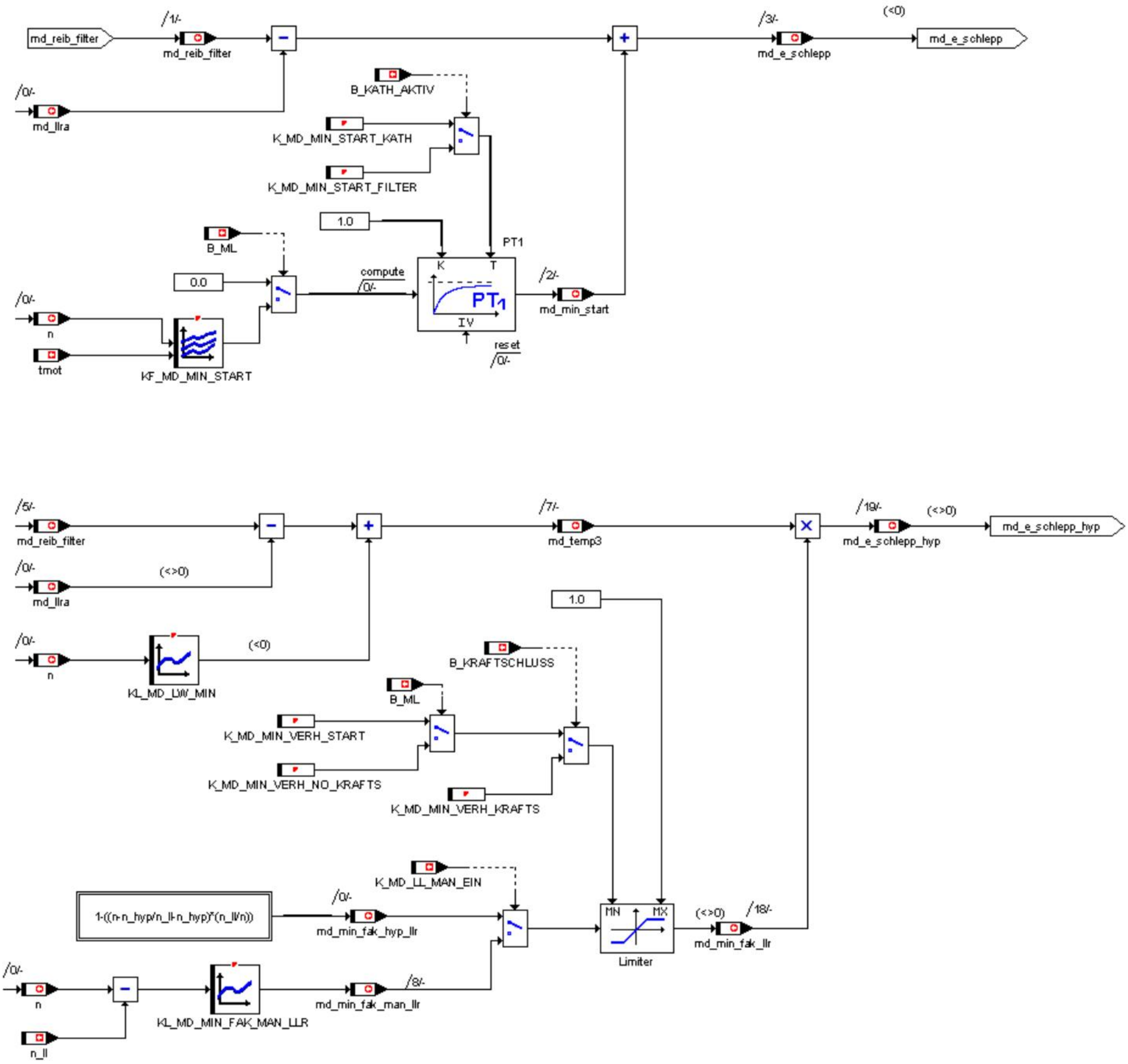
n_hyp = lfr_nsoll + KL_MD_MIN_DN_HYP(tmot)

The weighting factor is calculated as follows:

md_min_fak_hyp_llr = 1 - [(n - n_hyp) / (n_ll - n_hyp) * (n_ll / n)]

The slope of the hyperbola can be influenced with the "KL_MD_MIN_DN_HYP" characteristic. High values mean a flat hyperbola, low values mean a steep hyperbola. The torque can also be changed manually using the "KL_MD_MIN_FAK_MAN_LLR" characteristic, in order to be able to influence the torque gradient in the idling speed range with great flexibility, for example.

Image: Calculation of drag torque (md_schlepp.gif)



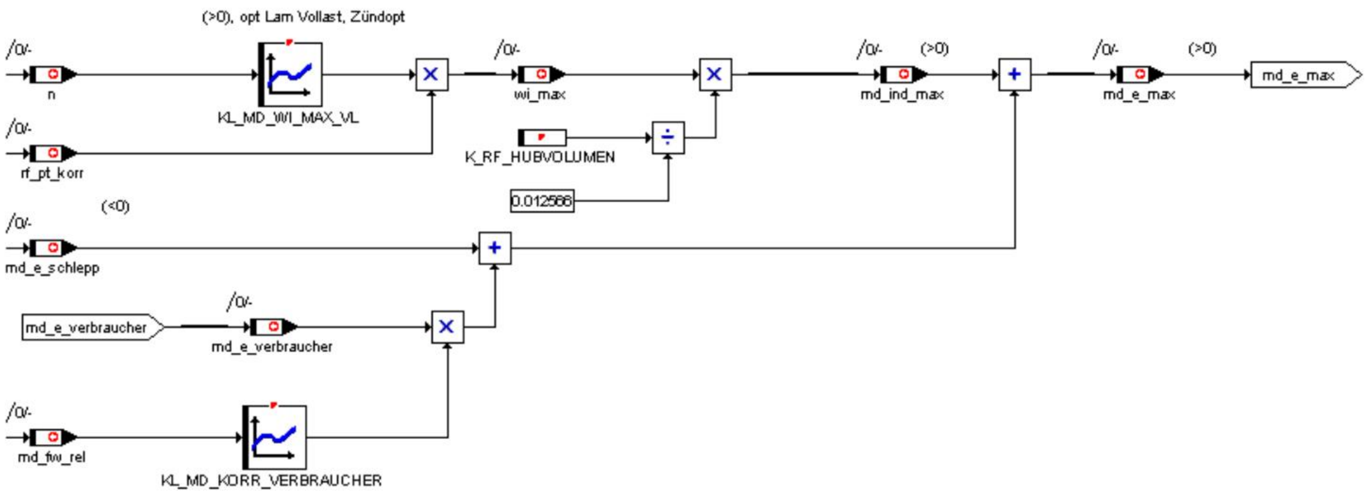
6. MAXIMUM INDICATED TORQUE

The maximum "wi" that the engine can achieve in full-load operation and under standard conditions, full-load lambda and optimum ignition at the respective speed is stored in the "KL_MD_WI_MAX_VL" characteristic curve. By correcting for the real environmental conditions, the currently possible maximum "wi_max" is obtained.

The indicated moment is calculated from the indicated work. With this, the drag torque including consumer "md_e_schlepp" and "md_e_verbraucher" is added, so that the effective maximum torque "md_e_max" is created. The addition of the consumption moments is weighted via the characteristic "KL_MD_KORR_VERBRAUCHER"=f(md_fw_rel), small accelerator pedal positions of up to approx. 80%, the consumers are not included. Only then will a full calculation be made at the pedal travel of up to 100%. This avoids

"Md_e_ Verbraucher" is the input variable that takes into account all consumers as well as the KSG torque, which, among other things, supplies the power for the valve control.

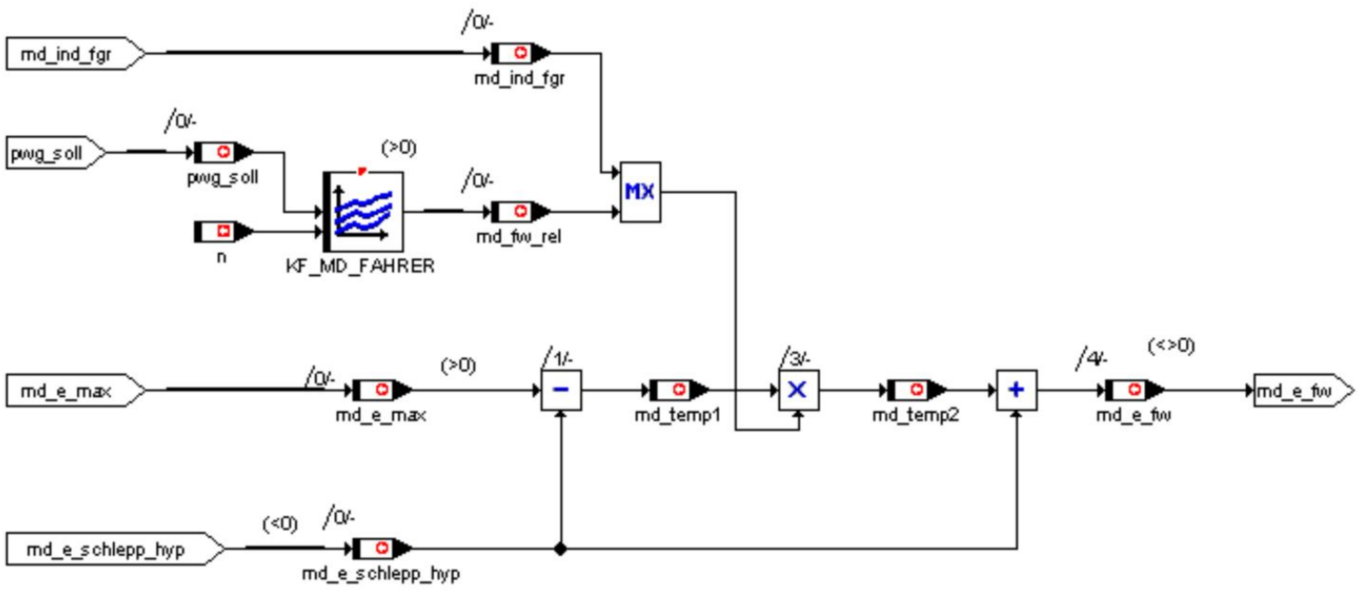
Image: Calculation of the maximum effective moment (md_max.gif)



7. CALCULATION OF DESIRED MOMENT

The driver's request is determined by the PWG module in the form of a relative pedal position "pwg_soll", with 0% corresponding to a non-actuated accelerator pedal and 100% corresponding to the full-load stop of the pedal. This relative pedal position is converted into a relative driver request "md_fw_rel" via the "KF_MD_FAHRER" map, which in turn is between 0 and 100%. 100% corresponds to the maximum effective torque "md_e_max". 0% corresponds to the drag torque with hyperbola increase "md_e_schlepp_hyp" (without consumers).
At the same time, a relative torque request “md_ind_fgr” can also be determined by the vehicle speed controller module.
The effective driver's desired torque "md_e_fw" is then determined by adding "md_e_schlepp_hyp".

Image: Calculation of the driver/FGR desired torque (md_fw.gif)



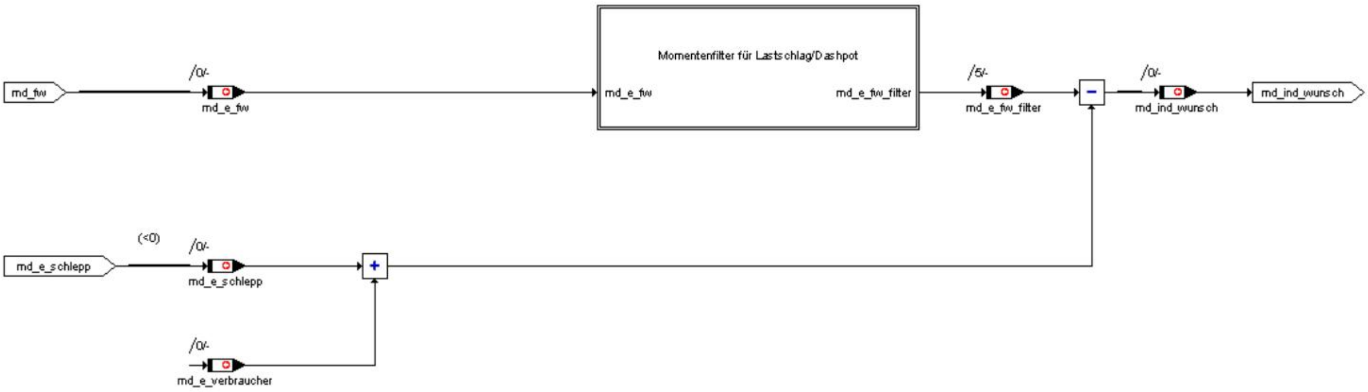
8. IMMEDIATE FILTER

The task of the dynamic filter is to filter the torque requirements in certain states and thus limit the gradients.

The load impact/dashpot filter takes over the filtering of the driver's desired torque "md_e_fw". Depending on the operating point, a maximum permitted positive (load impact) or negative (dashpot) Torque change per time unit is calculated and the torque requirements from the driver or FGR constrained to these gradients.

The filtered driver's desired torque "md_e_fw_filter" is then subtracted from the drag torque "md_e_schlepp" and the consumer torque "md_e_ Verbraucher" and saved in "md_ind_ Wunsch".

Image: Torque filter overview (md_filter.gif)



8.1. DYNAMIC FILTER FOR DESIRED TORQUE GRADIENTS

For damping large positive (load impact) and large negative (dashpot) torque gradient, the dynamic filter becomes active.

Load impact and dashpot filters are structured in a similar way. In principle, they only differ in the direction of the moment gradient:

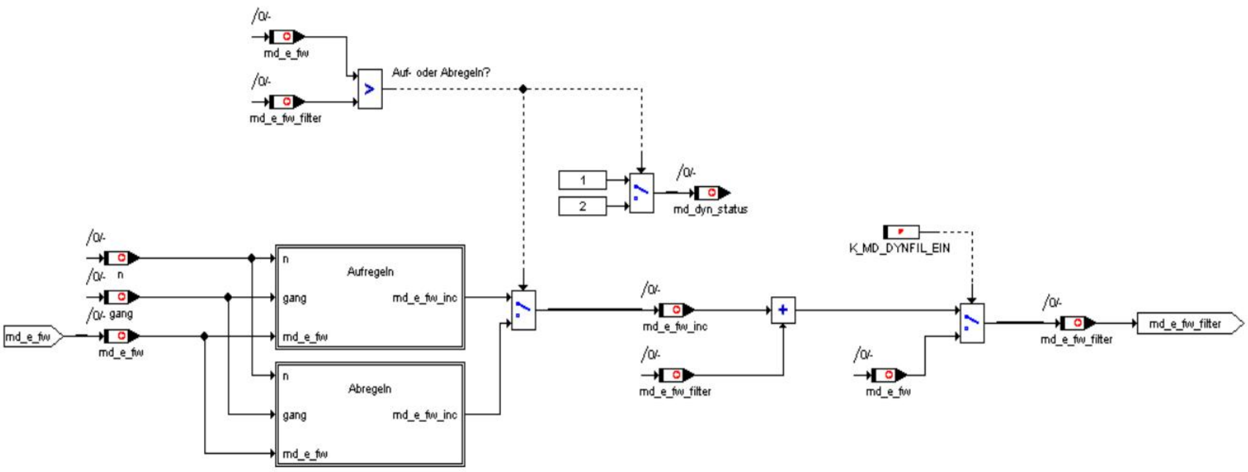
load stroke :	positive moment gradient
Dashpot :	negative moment gradient

The dynamic filter can be switched on and off using the application constant "K_MD_DYNFIL_EIN". If you drive stationary, the dynamic filter does not work (md_dyn_status = 0).

In the case of upregulation, there is an applicable threshold value "K_MD_DYNFIL_AUF_12" for the torque, at which there is a switchover from area 1 to area 2. If another threshold "K_MD_DYNFIL_AUF_23" is exceeded, the system switches to area 3.
The time-dependent increment is taken from a range-dependent engine speed and gear-dependent map. It should be noted that flatter ramps are applied in area 2 in order to minimize motor stalling in the area of effective torque=0. Steeper ramps are applied in areas 1 and 3.

There are three areas and two threshold values "K_MD_DYNFIL_AB_12" and "K_MD_DYNFIL_AB_23" for the case of curtailment.

Image: Dynamic filter overview (md_filter_dyn.gif)



The torque increment "md_e_fw_inc" by which the new filtered driver request is increased is calculated as follows:

md_e_fw_inc = (md_limit_r - md_limit_l) / (map value from KF_DYNFIL_AUF/AB)

md_e_fw_filter = md_e_fw_filter_alt + md_e_fw_inc

The ramp gradients can be influenced in the individual areas via the characteristic fields "KF_MD_DYNFIL_AUF/AB1..3". The higher the applied values of the characteristic maps, the flatter the filter regulates the torque up or down.

Table : Ramp gradients

	Area	**target value md_limit_r**	**old value md_border_l**
turn up	1	md_e_fw	md_e_fw_filter_alt
turn up	2	K_MD_AUF23	K_MD_AUF12
turn up	3	md_e_fw	md_e_fw_filter_alt
limit	3	md_e_fw	md_e_fw_filter_alt
limit	2	K_MD_AB12	K_MD_AB23
limit	1	md_e_fw	md_e_fw_filter_alt

Activation of the filter

If the filtered output value "md_e_fw_filter" is not equal to the input value "md_e_fw", the filter is activated. If "md_e_fw" > "md_e_fw_filter" then regulation/load impact is recognized and the bit "md_dyn_status" is set to the value 1. If "md_e_fw" < "md_e_fw_filter", curtailment/dashpot is recognized and the "md_dyn_status" bit is set to the value 2.

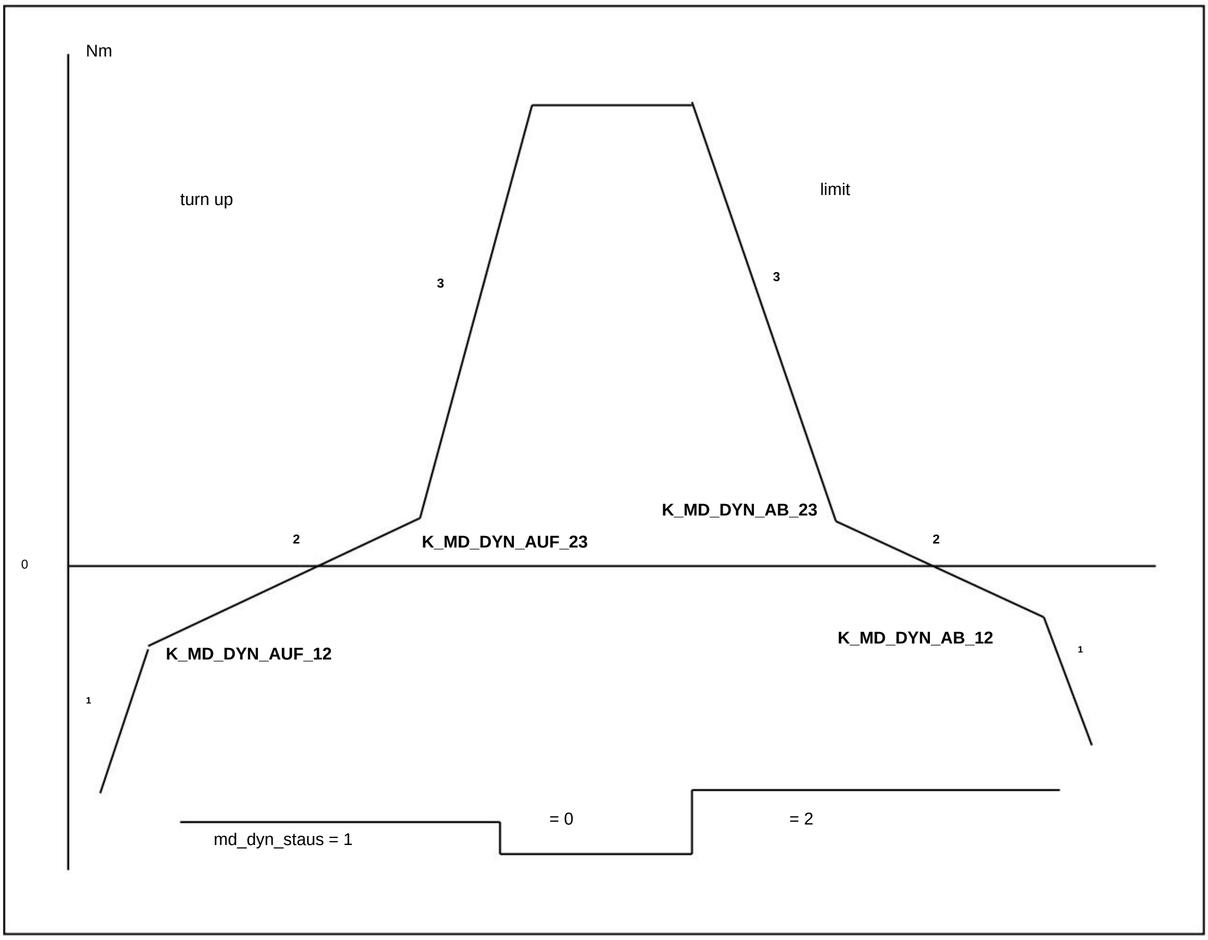
Deactivation of ramping up to stationary operation

If the filtered output value "md_e_fw_filter" exceeds the input value "md_e_fw", the filter is deactivated. The output is equated directly with the input ("md_e_fw_filter" = "md_e_fw") and the "md_dyn_status" bit is reset to the value 0.

Deactivation from derating to stationary operation

If the filtered output value "md_e_fw_filter" falls below the input value "md_e_fw", the filter is deactivated. The output is directly set equal to the input ("md_e_fw_filter" = "md_e_fw") and the "md_dyn_status" bit is reset to the value 0.

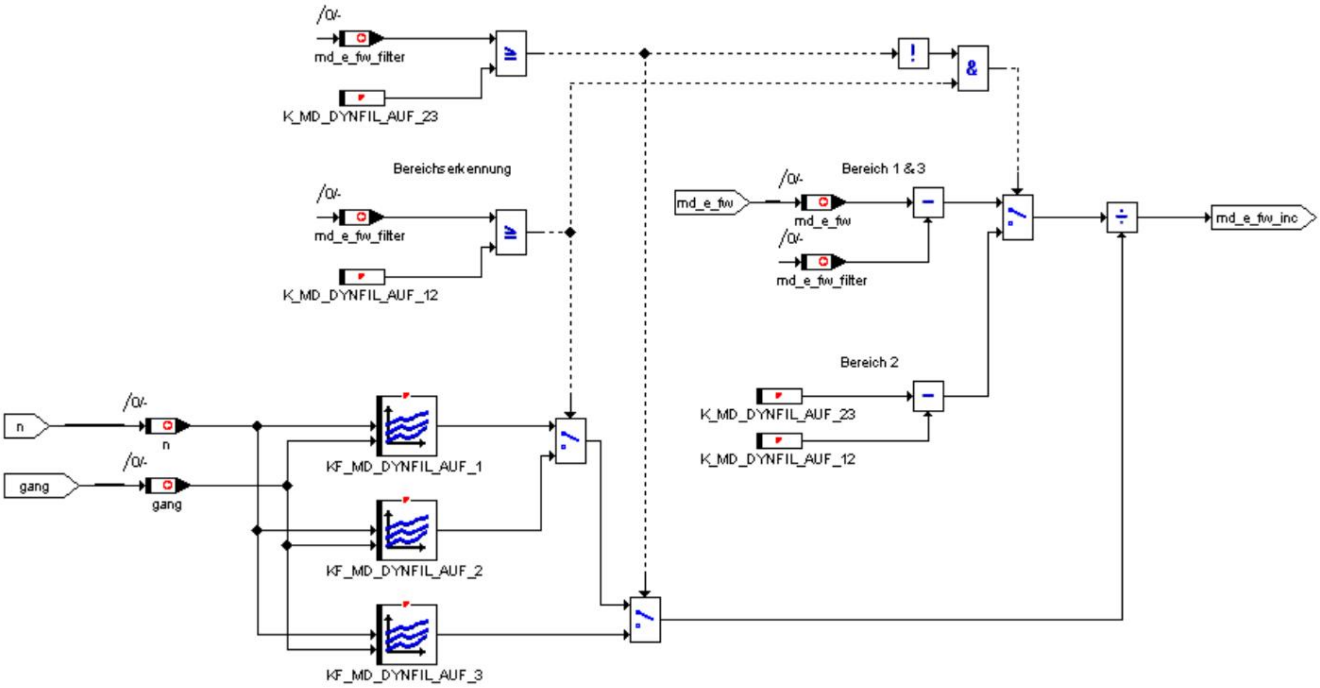
Image: Classification of the dynamic filter areas



Machine Translated by Google

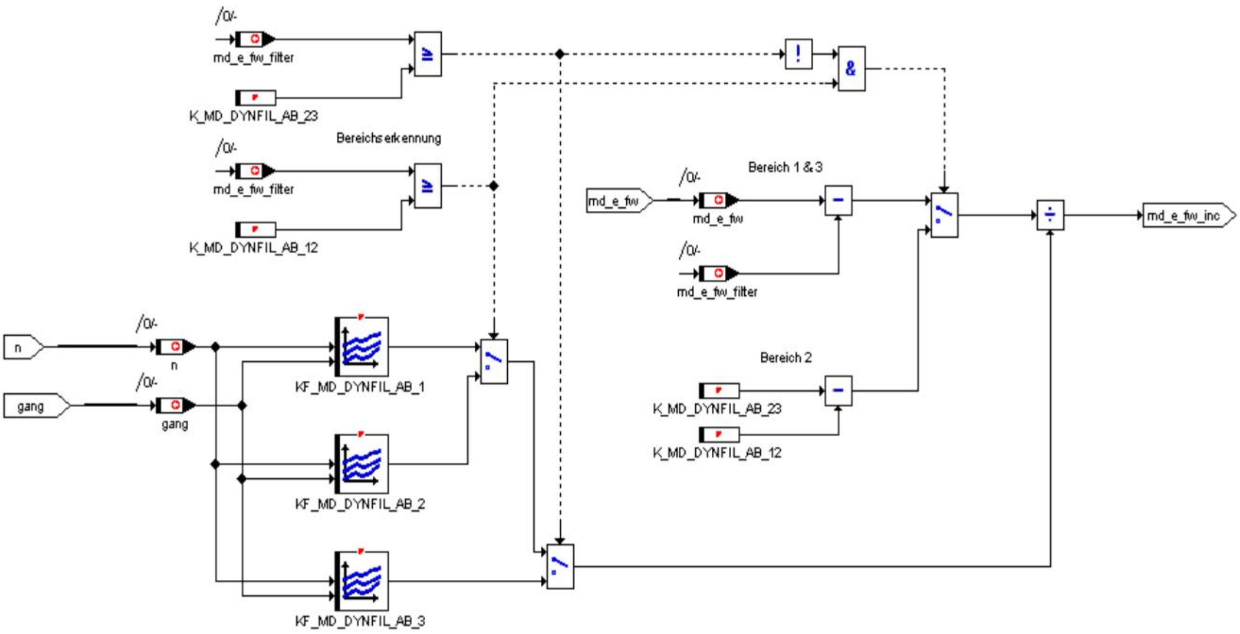
8.1.1. LOAD IMPACT FILTER

Image: Load impact operation of the dynamic filter (md_filter_ls.gif)



8.1.2. DASHPOTFILTER

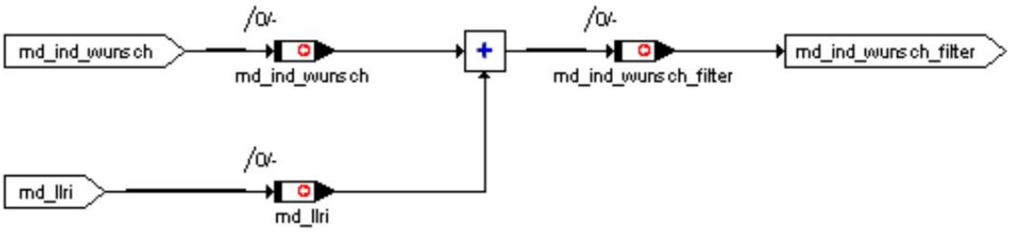
Image: Dynamic filter dashpot operation (md_filter_dashpot.gif)



8.2. INTERVENTION OF THE IDLE REGULATOR

The torque component of the idle controller is calculated as shown in the following figure.

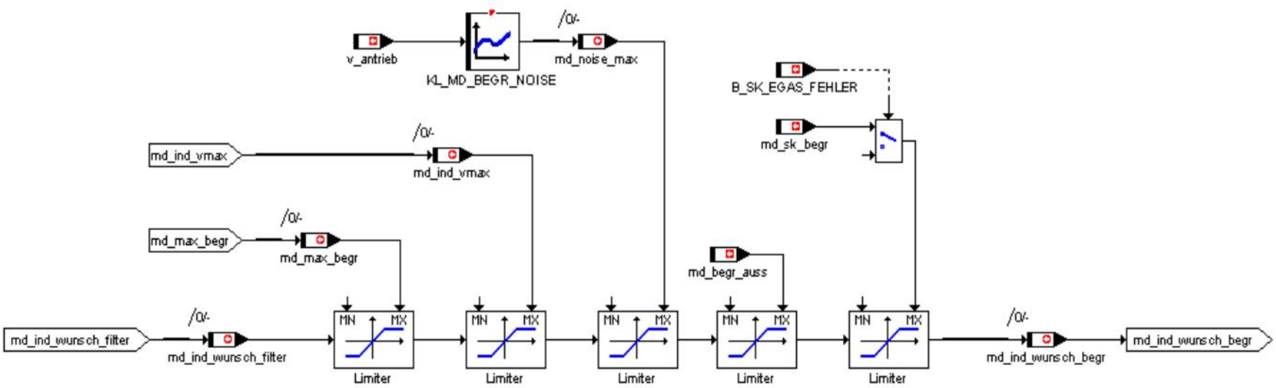
Image: Calculation of the idle controller component (md_eingrlfr.gif))



9. TORQUE LIMITATIONS

An overview of the torque-limiting interventions is shown in the figure below. The individual limitations are described in the following subsections.

Image: Overview of torque limitation (md_limitation.gif))



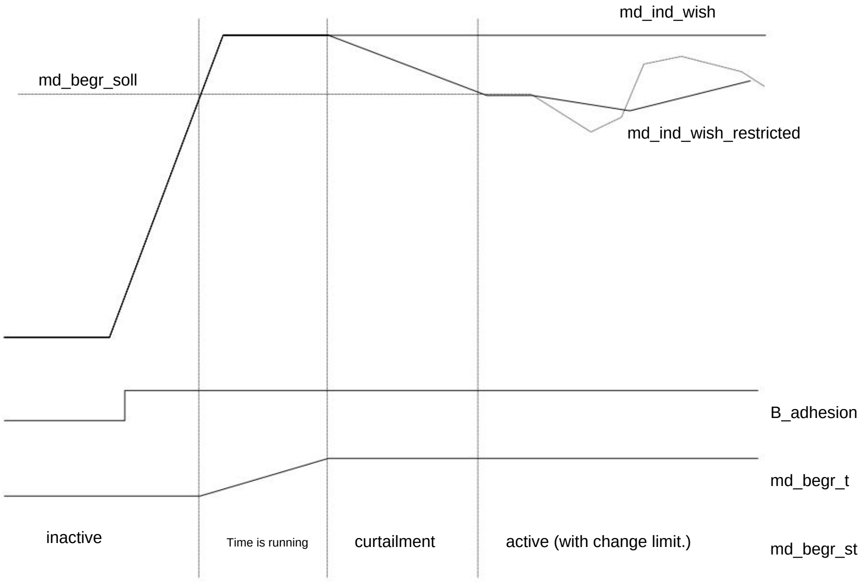
9.1. TORQUE LIMITATION

The engine torque (torque at the clutch) must be limited due to the transmission's torque strength being too low.

The maximum permissible indicated engine torque is calculated from the "KL_MD_BEGR_GANG" characteristic curve, which contains the gear-dependent maximum torques, plus the internal engine torque losses "md_ind_schlepp". If the speed gradient is positive, there is a further correction for the motor moment of inertia. The influence of an ignition angle retardation of the knock control or knock adaptation is taken into account via the ignition angle efficiency "md_eta_zw_ve".

md_max_begr = (KL_MD_BEGR_GANG (gang)
+ md_ind_haul
+ K_MD_J_MOTOR * d_n40 falls d_n40 > 0
) / md_eta_zw_ve

The torque limitation is only intended to prevent continuous operation of the motor above the maximum torque. Exceeding the torque limit for a short time, such as during acceleration measurements, is not considered critical for the transmission. The functionality of the torque limitation is adapted to this. After each interruption of adhesion, the torque limitation becomes inactive for the period "K_MD_BEGR_T", whereby the time only starts to run when the maximum threshold is exceeded for the first time. Then, based on the current desired torque, the limitation threshold is reduced to the target value "md_max_begr" via the "K_MD_BEGR_RAMPE" ramp. After the end of the regulation, changes of "md_max_begr", which can be very fast due to speed gradients and KR/KA influences, are limited via the change limitation "K_MD_BEGR_DELTA".



9.2. SPEED LIMIT

With the MSS60, the Vmax limitation acts directly on the EGas system via the torque manager. The maximum speed is controlled via an I controller in two stages.

Vmax ready: v > vmax_ready

When "vmax_berei" is exceeded for the first time, which must be below the Vmax "K_V_MAX", a torque is calculated in advance, which should enable stationary driving at the Vmax point.

To do this, the current indicated torque is corrected by the excess torque, which currently causes a vehicle acceleration and then multiplied by the square of the quotient of maximum speed and actual speed, since the air resistance also increases quadratically with the vehicle speed.

md_ind_vmax = (md_ind_ Wunsch_red_korr - K_MD_J_FZ * d_v) * (K_V_MAX / v)2

Vmax regulation: v > K_V_MAX

When K_V_MAX is exceeded, an I controller becomes active, which integrates the maximum permissible torque "md_max_begr" up or down according to the controller deviation.

md_ind_vmax = md_ind_vmax + K_MD_I_VMAX * (K_V_MAX - v)

Since, due to the I controller, "md_ind_vmax" can be very small or even overflow, "md_ind_vmax" is limited to the values K_MD_VMAX_MIN or K_MD_VMAX_MAX.

The Vmax control is deactivated again as soon as the vehicle speed has fallen below the threshold K_V_MAX - K_V_MAX_HYS.

"md_ind_vmax" in the torque manager is only taken into account with active Vmax limitation, not with Vmax readiness.

9.3. TORQUES FOR CATS DAMAGE MISS- OUTS

In the event of cat-damaging misfires, a torque limiter becomes active, which is intended to reduce the charge and thus the air flow through the catalytic converter depending on the current engine speed.

The torque limitation is activated as soon as a cylinder has to be switched off due to misfires that could damage the cat.

Phase 1: Expiration of the waiting time "K_MD_BEGR_AUSS_TIME", in which no torque limitations are effective to prevent any critical driving situations.
Phase 2: Ramp-shaped curtailment with "K_MD_BEGR_AUSS_ABREG", based on the driver's request "md_ind_ Wunsch" to the limiting torque.
Phase 3: Torque limitation active.

Calculation of the limiting torque:
md_begr_auss = KL_MD_BEGR_AUSS = f (n)

The current state of the torque limitation is visible in the "md_begr_auss_st" variable.

The torque limitation remains **active until the engine is switched off,** even if no further misfires have been detected in the meantime.

9.4. FUEL PRESSURE COLLAPSE TORQUES

If the fuel pressure collapses and the tank is empty at the same time, a torque limiter becomes active which, depending on the current engine speed, is intended to reduce the filling and thus the air flow through the catalytic converter.

The torque limitation is activated as soon as the cat protection function sets the activation release based on the four lambda probe signals and the tank level.

Since the engine is no longer running at this point due to the collapsed fuel pressure, the torque is limited immediately and without derating.

Calculation of the limiting torque:
md_begr_auss = KL_MD_BEGR_FST = f (n)

The current state of the torque limitation is also visible in the "md_begr_auss_st" variable.

The torque limitation remains **active until the engine is switched off.**

9.5. TORQUE LIMITATION FOR NOISE REDUCTION

To minimize the noise, a function is implemented which, after setting the noise reduction condition (for more information, see module description: gang.doc), depending on the current vehicle speed, limits the maximum indicated torque of the engine.

md_noise_max = KL_MD_BEGR_NOISE = f (v_antrieb)

10. MOMENTENRESERVE

Since the current ignition angles generally correspond to the optimum ignition angle value, only torque reductions are possible via ignition angle interventions. However, in certain operating ranges, such as idling speed control, it is desirable to be able to quickly build up torque via an ignition angle intervention.

For this purpose, the torque specification for the charge path is increased via the torque reserve module, while the torque specification for the ignition angle path remains unchanged. This leads to the fact that the filling and thus the actual torque before intervention increases. Thus, the actual torque before intervention exceeds the torque requirement of the ignition angle path and the excess torque is compensated for by retarding the ignition angle. This compensated excess torque is now available for a quick increase in torque by advancing the ignition angle.

md_res_kath: Momentenreserve der Katheizfunktion

md_res = md_res_kath

10.1. TORQUE RESERVE FOR KATHEIZ FUNCTION

In the case of the catalytic converter heating function, the torque reserve is used to reduce efficiency and thus increase exhaust gas temperatures.

For this purpose, an offset torque is calculated as a function of the operating status, engine speed, load, engine temperature and time since the start, which is added to the desired torque for the filling path and compensated again by an ignition angle intervention.

The offset torque is made up as follows:

md_res_kath =	KF_MD_RES_KATH	Offsetmoment = f(n, wi)
	* KF_MD_RES_KATH_GEW	weighting factor = f(tmot, t_ml)
	* md_res_kath_faktor	Weighting factor increase/decrease

The determination of the weighting factor "md_res_kath_faktor" itself can be divided into five areas:

Area 1: Start or post-start (until the start torque limitation ends)
weighting factor = 0

Area 2: Adjusting the weighting factor
the weighting factor becomes linear with the increment from the start value "K_MD_RES_KATH_T_AUFREG" raised to the value 1.0

Area 3: Torque reserve for cat heating fully active
weight factor = 1.0

Area 4: Limiting the weighting factor

After removing the condition "B_KATH_AKTIV_MDRES", the weighting factor is reduced linearly to zero with the increment "K_MD_RES_KATH_T_ABREG".

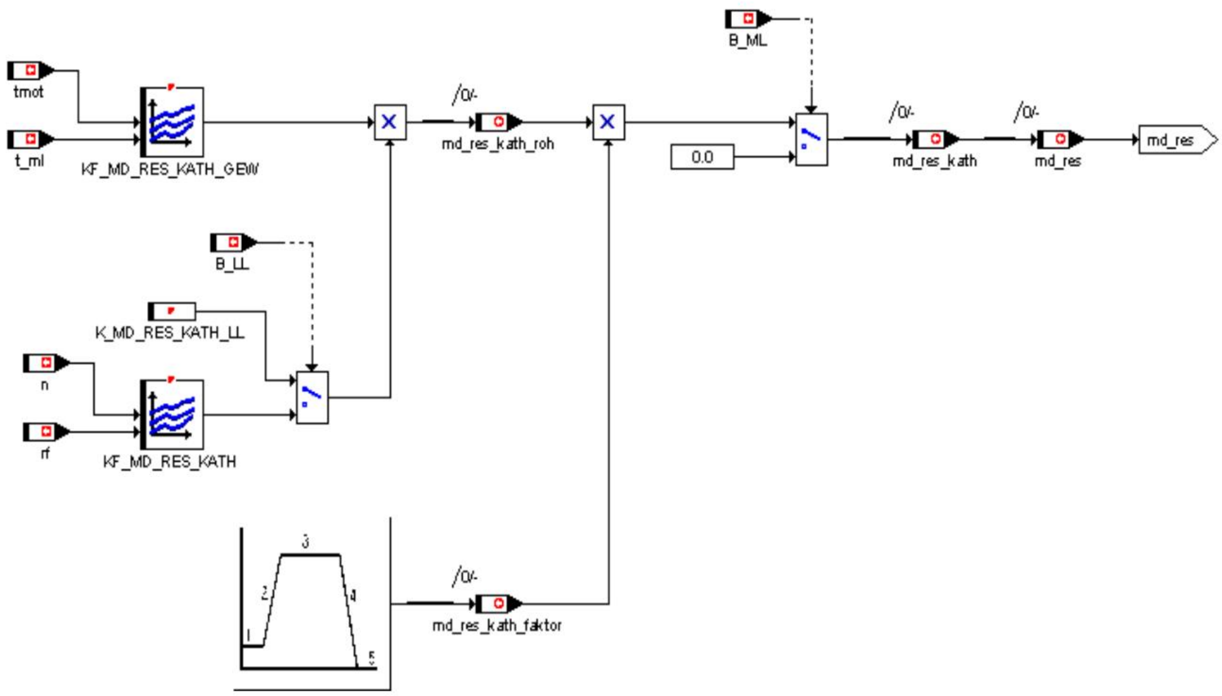
Area 5: Torque reserve for cat heating inactive

weighting factor = 0

The activation condition for the torque reserve for catalytic converter heating is identical to the activation condition for the ignition angle intervention for catalytic converter heating.

The torque intervention for cat heating can be blocked via the constant "K_MD_RES_CONTROL".

Image: Overview of torque reserve for catalytic converter heating (md_reservekath.gif))



10.2. TORQUE RESERVE FOR STRONG STEERING ENCLOSURES (NOT IMPLEMENTED IN EVT !)

When the steering reaches its end stop, the power steering servo pump absorbs a lot of torque, which can cause the engine speed to drop when idling and possibly even cause the engine to stall. The response of the idling controller to an increase in charge is too slow due to the gas transit times. Therefore, depending on the steering wheel angle, a torque reserve should be built up in advance, which then allows a rapid increase in torque by advancing the ignition angle if the idle speed falls below the target value.

Activation condition: v < K_MD_RES_LRW_V
Deactivation: v > K_MD_RES_LRW_V + K_MD_RES_LRW_VHYS

Calculation algorithm in bullet points:

- Calculation of the amount of the steering angle
- Calculation of the raw value of the torque reserve "md_res_lrw_loc" via the characteristic KL_MD_RES_LRW = f(lrw_abs)
- Change limitation of the torque reserve to "K_MD_RES_LRW_DELTA" resulting torque reserve: md_res_lrw_roh
- Consider a possibly already existing ZW retard from the Katheizmodul

  md_res_lrw = md_res_lrw_raw - (md_ind_ Wunsch_begr * md_eta_kath_offset)

  The term md_ind_ Wunsch_begr * md_eta_kath_offset is the torque in Nm, which is already available as a torque reserve due to the ZW late drawing of the catalytic converter heating function. A torque reserve for steering assistance that may be present at the same time therefore only has to take the delta into account.

The status of the torque reserve is visible in the "md_res_lrw_st" variable:
Bit 0: Activation conditions met
Bit 1: Intervention active, ie intervention torque not equal to zero

10.3. TORQUE RESERVE LIMITATION

The working principle of the torque reserve presupposes that the increase in torque in the charge path can be compensated for by an ignition angle intervention. To do this, it must be determined how much leeway is still available for a torque reserve at the current operating point.

The remaining scope for retarding the ignition angle is the difference between the current ("md_eta_zw_ve") and the minimum possible ("md_eta_zw_min") ZW efficiency.

The factor that is still possible for increasing the filling specification is calculated as follows:

„Md_eta_res“ = 1 / (1 - (md_eta_zw_ve - md_eta_zw_min))

If the requested torque reserve exceeds this margin, the torque request is limited to the value "md_ind_ Wunsch_begr * md_eta_res".

11. MOMENTARY INTERVENTION FILLING PATH

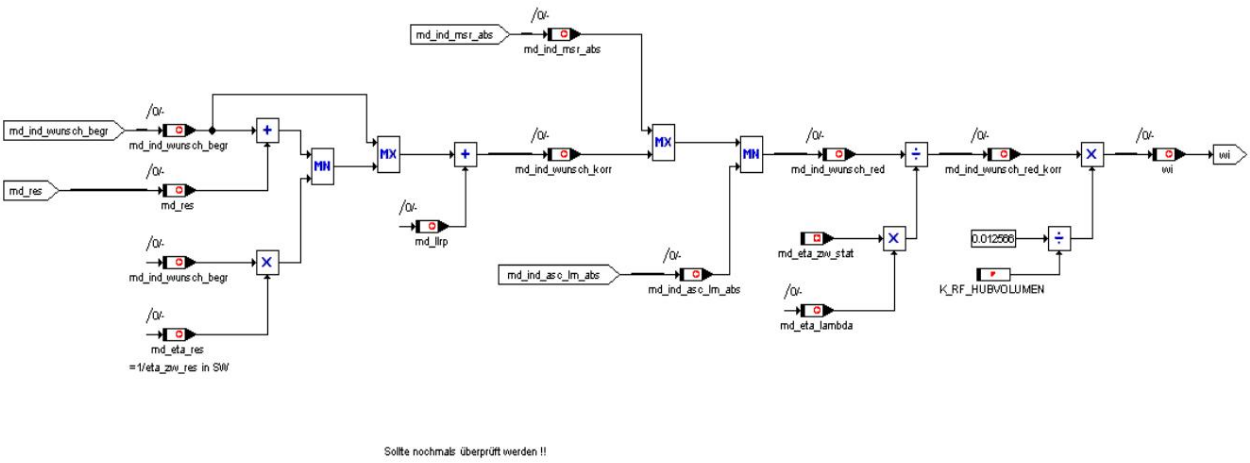
In this module, the torque is coordinated with the DSC system and with the other torque-reducing modules.

The system has two different intervention options. In the event of an MSR intervention (engine drag torque control), the DSC requests an increase in torque, which is set purely via the filling. In the event of an ASC intervention (Automatic Stability Control), the ASC system can request torque reductions separately for the filling and ignition angle paths.

The plausibility check of the DSC interface and the implementation of the requirements in indicated moments is described in the "CAN interface" chapter, so that only the two intervention moments "md_ind_asc_lm_abs" and "md_ind_msr_abs" are considered here.

The requested torque interventions are taken into account via max and min selections with regard to the desired torque “md_ind_wish_red”. The order of selection is described in the graphic below.

Image: Moment interventions in filling path (md_fuellung.gif)



11.1. EFFICIENCY CORRECTION

A big benefit of the torque manager is that it can be used to compensate for actions by other modules that affect the torque, such as lean warm-up or catalyst heating

The influence of a stationary ignition angle retard and a deliberate deterioration in efficiency for the catalyst heating function in the ignition angle efficiency "md_eta_zw_stat" is calculated and the torque loss caused by this is compensated for at this point by a charge correction.

Likewise, the influence of the lambda value on the torque output is recorded in a lambda efficiency "md_eta_lambda" and compensated for by a filling correction. However, only lambda efficiency values lower than "K_MD_ETA_LAMBDA_MAX" are taken into account.
Efficiencies greater than this value are limited to this value.

The new torque requirement is composed as follows:

md_ind_wish_red_corr = md_ind_wish_red /md_eta_zw_stat / md_eta_lambda

The efficiency correction can also be deactivated for application purposes via the "K_MD_ETA_MCS" constant.

Bit 0 = 1 :	Correction via Lambda active
Bit 1 = 1 :	Correction via stationary ignition angle including catalytic converter efficiency active
Bit 7 = 1 :	Correction only active via cat heating efficiency, but not via stationary angle

11.2. CALCULATION OF WI

The specific, indexed work "wi" is calculated from the indexed, corrected desired moment "md_ind_ Wunsch_red_korr". "Wi" is used as an input variable for various maps (e.g. control edges) and has the advantage that it contains the displacement and is therefore independent of the displacement variant.

The associated formula is:

$$M_d = \frac{V p_{mi}}{4 ÿ} \quad (1)$$

or numerical value equation : $M_d = \frac{V p_{mi}}{0,12566}$ (2) Md[Nm],V [dm^3] ,p$_{me}$ [bar]

myth in $= \frac{1}{10} p_{me}$ (3) wi[kJ/dm^3] ,p$_{me}$ [bar]

olgt:

$$w = md\_ind\_wish\_red\_corr \cdot \frac{1}{K\_RF\_HUBVOLUMEN} \cdot 0,012566$$

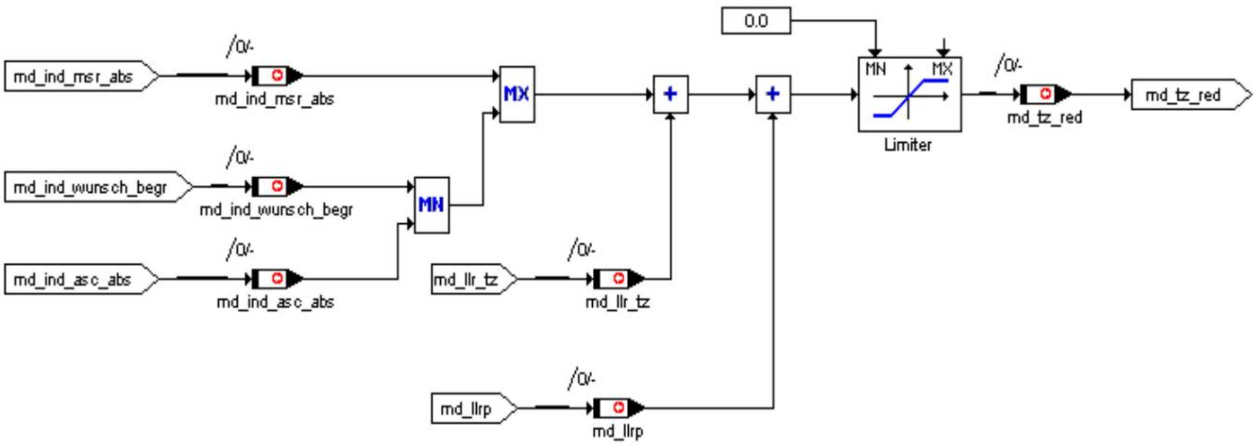
12. CALCULATION OF CONTROL EDGES

In the EVT engine, the filling is not realized by the throttle valve angle, but by the control edges. In this Module (Evt_momentenrealisierungs.doc and Betriebsartenmanager.doc) the target throttle valve angle for 50 mbar vacuum for tank ventilation, the basic ignition angle, the target air mass flow and the advance angle are calculated in addition to the control edges.

13. TORQUE INTERVENTION IGNITION PATH

The DSC torque intervention in the ignition angle path runs in the same way as the charge path via a maximum value (MSR function) or a minimum value selection (ASC function). The idling control also has a torque intervention "md_llr_tz + md_llrp", which only acts on the ZW path of the torque manager and which can counteract torque-reducing measures of other modules. The output variable "md_tz_red" is limited to positive moments.

Image: DSC and LLR torque interventions in the ignition angle path (dsc_llr_mdeingriff_zw.gif)



14. CALCULATION OF TIMING ANGLE INTERVENTION

In this module, the torque requested for the ignition angle path is converted into an intervention ignition angle, taking into account the actual ignition angles of the individual cylinders.

The conversion takes place cylinder-selectively once per working cycle about 360 degrees before the firing TDC of the corresponding cylinder. This is to guarantee that on the one hand the ignition angle interventions of the knock control are already available, on the other hand there is still enough time to update the ignition channel before the ignition coil is energized.

14.1. CALCULATION OF OPTIMUM FIRING ANGLE

The optimum ignition angle zw_opt is the ignition angle at which the ignition hook has its apex, ie the indicated engine torque/work reaches its maximum value under standard conditions. The theoretically optimum ignition angle can be earlier than the ignition angle that can be used at the corresponding operating point. This ignition angle is the reference for calculating the ignition angle interventions in the torque manager.

The calculation of the indicated actual and target wi is related to this optimal ignition angle.
A map for the optimum ignition angle must be available for each EVT operating mode.

(1) zw_opt = KF_ZW_OPT_x(n, wi)
with x = 0, 1, 2, 3, 4, 5, 7, 11, 12 (operating modes bm_evt_state)

The influence of the engine temperature and the lambda value is also taken into account in the corrected optimal ignition angle zw_opt_korr.

(2) zw_opt_korr = zw_opt + KF_ZW_TMOT(tmot,wi) + zw_lambda

Machine Translated by Google

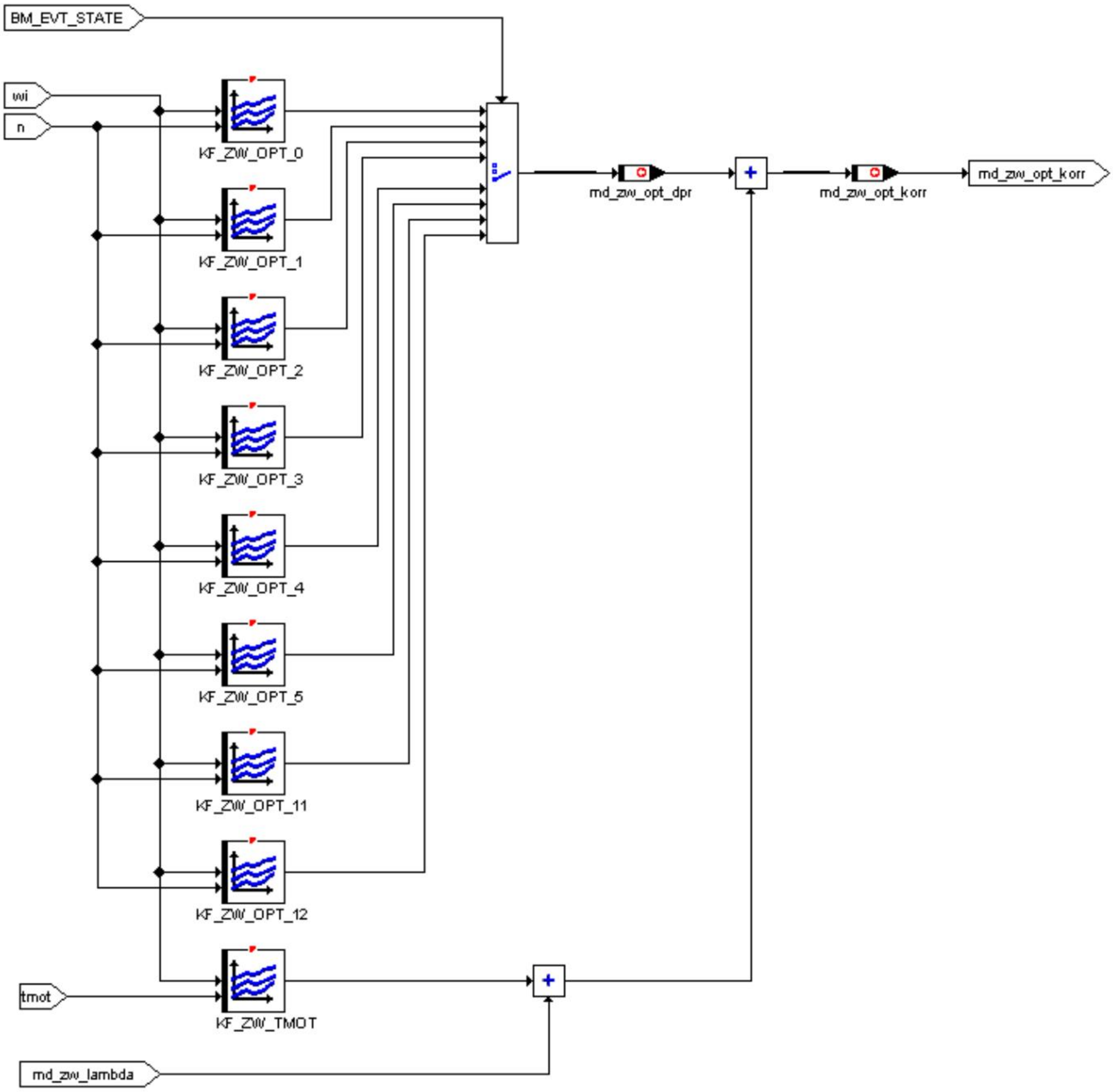


Image: Calculation of optimal ZW (ZW_Eingriff2_2.gif)

14.2. CALCULATION OF FIRING ANGLE EFFICIENCY

The ignition angle efficiencies are required in order to take existing ignition angle interventions into account when calculating the actual wi. Furthermore, the available adjustment range for ignition angle interventions of the torque manager is calculated via the ignition angle efficiencies.

14.2.1. MINIMUM FIRING ANGLE EFFICIENCY

The variable "md_eta_zw_min" includes the efficiency that can be achieved with the latest permitted ignition point "tz_min". Wi reductions below this level of efficiency cannot be fully represented by interventions in the ignition angle:

(1) md_eta_zw_min = Fkt.(Zündhaken_Polynom(tz_min))

14.2.2. BASIC **FIRING ANGLE EFFICIENCY**

The variable "eta_zw_bas" includes the efficiency that is achieved with the current, corrected base ignition angle "tz_bas_korr". The corrected basic ignition angle is made up of the basic ignition angle maps of the operating modes, a correction via engine temperature and an ignition angle offset through lambda variation.

(2) tz_bas_korr = tz_bas + tz_tkorr + md_zw_lambda

The basic ignition angle tz_bas is calculated in the torque realization module.

With active catalyst heating, the basic efficiency is reduced by the amount "eta_zw_kath_offset".

(3) eta_zw_bas = Fkt.(Zündhaken_Polynom(tz_bas_korr)) - eta_zw_kath_offset

14.2.3. **CALCULATION** OF FIRING ANGLE **EFFICIENCY BEFORE INTERVENTION**

The variables "eta_veX" are based on the calculated ignition angles before intervention "tz_veX", which also contain adjustments from knock control, knock adaptation and dynamic derivative action. Since the individual ignition angles are cylinder-specific, the efficiency must also be calculated cylinder-specifically.

(4) eta_zw_ve [x] = Fkt.(Zündhaken_Polynom(tz_ve[x]))
with x = 1, ..., number of cylinders

(5) eta_zw_ve = Mean (md_eta_zw_ve[x])
with x = 1, ..., number of cylinders

14.2.4. **CALCULATION** OF FIRING ANGLE **EFFICIENCY AFTER INTERVENTION**

First, the segment-synchronous torque intervention of the anti-judder control "md_ar" is added to the desired torque "md_tz_red". These interventions can normally only have a negative effect. When the torque reserve is active, however, it can also build up torque.

A target efficiency “md_eta_zw_soll” is calculated from the new target torque by dividing it by a reference torque, which is the measure for the ignition angle intervention. However, the implementation for six and eight cylinders differs in the type of reference torque, so that the selection can be configured using the constant "K_MD_BEZUG_ZW".

K_MD_BEZUG_ZW = actual torque:
The actual torque calculated from the speed and relative charge is used as the reference torque for all ZW interventions.

K_MD_BEZUG_ZW = target torque:
In the case of active DSC interventions or Egas errors, the actual torque is also used as the reference torque. For all other interventions, the reference torque corresponds to the target torque specification for the filling path. This has the consequence that regardless of the

Positioning accuracy of the Egas system and all dynamic influences, only the difference between filling and ZW specification for the ZW intervention is relevant.

The value averaged over the cylinders:
(6) eta_zw_ne = Mean (md_eta_zw_ne[x])
with x = 1, ..., number of cylinders

In the case of ZW interventions due to DSC requirements or in the Egas emergency program, the number of currently hidden cylinders is also taken into account at the reference moment.

If there is a requirement to guarantee a minimum reduction in ZW efficiency for the cat heating function at all operating points, the calculated target efficiency is limited to the cat heating efficiency eta_kath.

The subsequent limitation to "eta_zw_min" ensures that the ignition angle interventions remain limited to the latest possible ignition angle.

A minimum required efficiency for the ignition angle intervention can be defined using the constant "K_ETA_EINGRIFF", ie an intervention is only carried out if it falls below this value. In the variable "st_tz" each bit represents a cylinder (bit 0 = cylinder 1), whereby a set bit means that an ignition angle intervention by the torque manager is currently active for this cylinder.

Finally, a plausibility check based on the ignition angle efficiency before intervention is carried out.

14.2.5. **CALCULATION** OF INTERFERENCE **ANGLE**

The calculated cylinder-selective ignition angle efficiency after intervention eta_zw_ne[x] must now be converted in a second stage into an absolute intervention angle related to the ignition TDC.

This is done using the ignition hook stored for this operating point, which indicates the ignition angle efficiency as a function of the angle difference to the optimum ignition angle.
In this case, however, there is an efficiency and an ignition angle is to be calculated, which is achieved by reversing the calculation algorithm. A spark parabola thus becomes an efficiency root.

Parabelgleichung: $Y = a x2 + b x + 1$

root equation: $x = \frac{ÿ\, ÿ bb\, 4a \sqrt{(1\, y)}^{2}}{2a}$

The result of the root equation is an offset ignition angle for the optimal ignition angle "zw_opt_korr" for which the ignition hook is set aside.

The switch at the end of the calculation can be used to decide when an intervention angle should be passed on to the ignition angle calculation. The following intervention conditions are hidden behind the B_ZÜNDEINGGRIFF switch:

ASC - intervention
or MSR - intervention
or (Egas emergency program level 3 or 4 active ?? Siko EVT??)
or Cat heating function active
or dynamic filter active due to SA/WE
or Dynamic filter active due to load impact / dashpot
or torque reserve active

or anti-judder control active
or idling stabilization via ignition angle active
or generally activated via K_MD_TZ_CONTROL

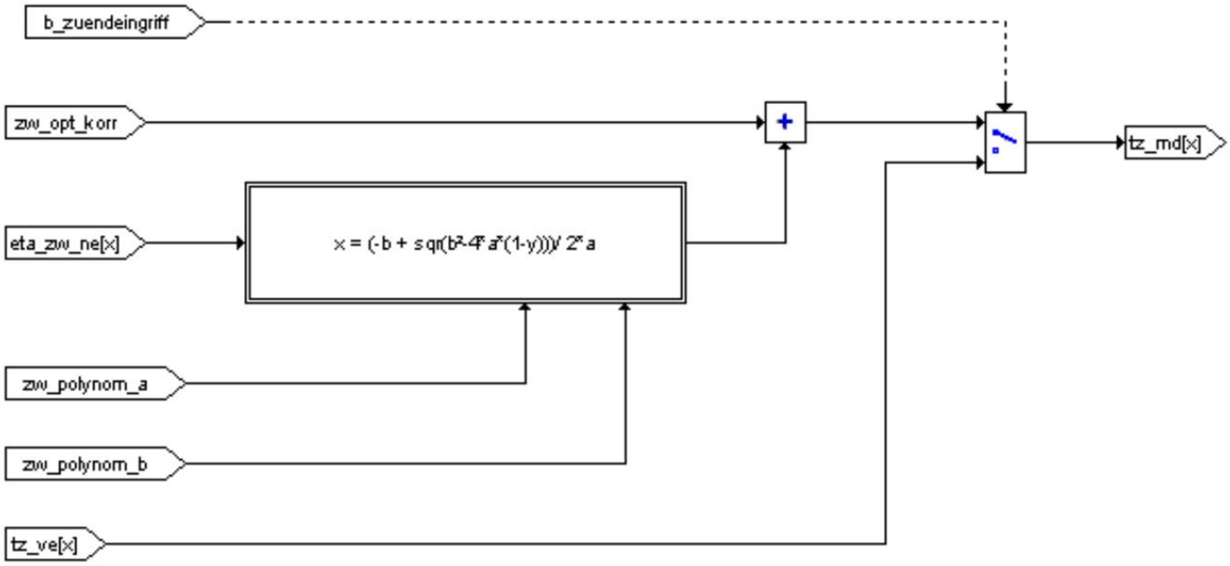


Image: Calculation of ignition angle efficiency after intervention (ZW_Eingriff_13_1.gif)

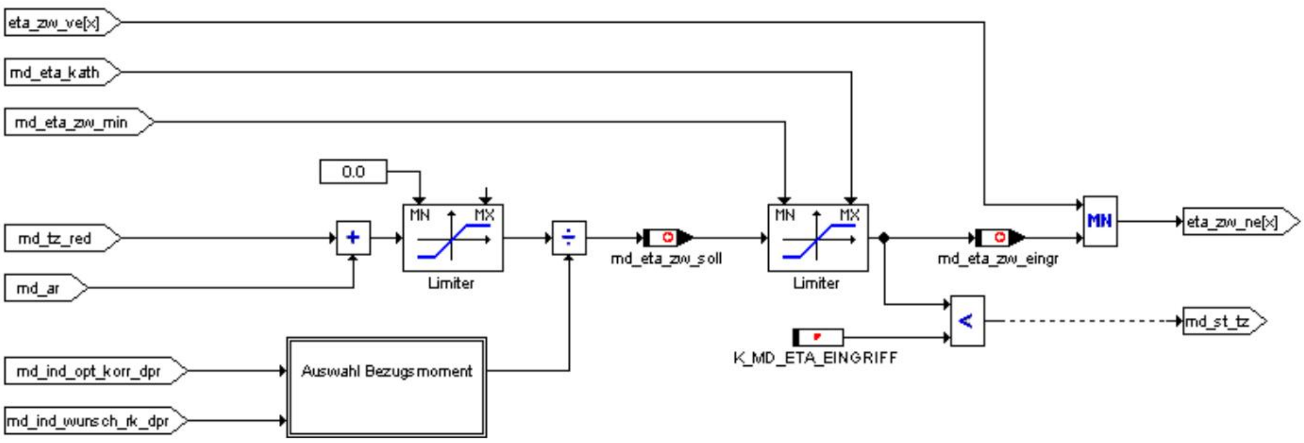


Image: Calculation of the intervention ignition angle (ZW_Eingriff_13_2.gif)

14.3. CALCULATION NORMALIZED PRIMER

The calculation of the ignition angle efficiencies and the intervention ignition angle of the torque manager is based on the normalized ignition hooks. For this purpose, speed and wi-dependent standard ignition hooks are stored.

The speed and wi-dependent standard ignition hooks are defined by the parameters a and b of the parabelgleichung

(1) y = ax2 + bx + 1

described.

The parabola always opens downwards, so the parameter a must always be negative. The input value x represents the difference between the optimal, corrected ignition angle and the ignition angle to be considered. Since the ignition angle output is limited to the minimum permitted ignition angle "zw_min", the ignition angles are also limited to this value within the ignition hook calculation.

The output value y represents the ignition angle efficiency, which can only reach the value 1.0 with an input value of zero (no infinitely flat ignition hooks). The output value is therefore always between the possible efficiency for "zw_min" and 1.0.

Substituting the general variables in equation (1) with the corresponding labels, it follows:

(2) eta_zw_x = zw_polynom_a * (dzw_x) ^{2} + zw_polynom_b * dzw_x + 1

Machine Translated by Google

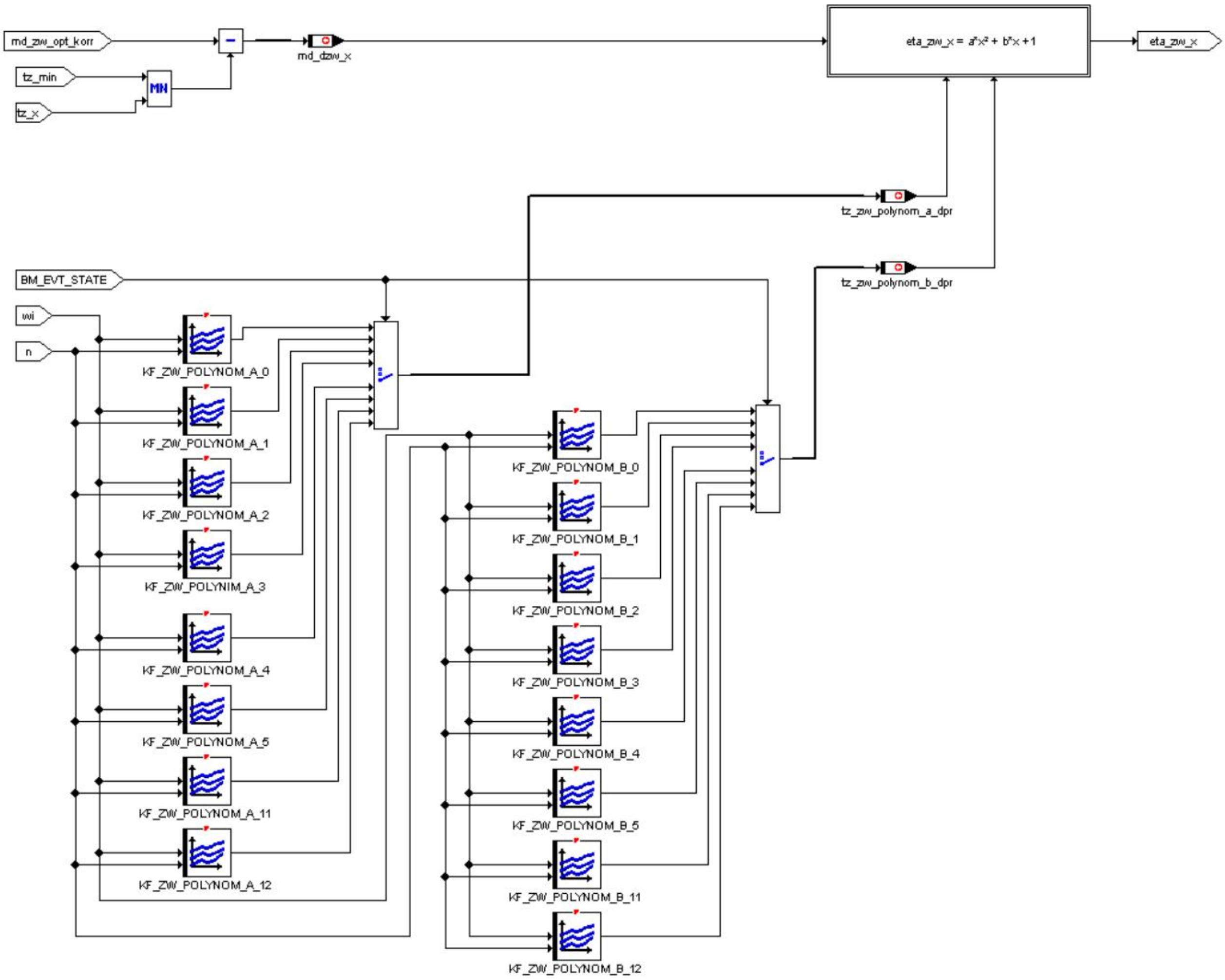


Image: Calculation of normalized ignition hooks (ZW_Eingriff2_3.gif)

15. MONITORING TORQUE CALCULATION

15.1. PROTECTION TORQUE CALCULATION

The main path of the torque calculation and all offset torques of other modules affecting it are checked for plausibility within the torque manager. If an implausible value is detected, this value is immediately converted into a neutral value and an error filter is started. After the error filtering has expired, the Egas monitoring function is notified, which then switches the Egas system to emergency operation level 2 - emergency driving via the idle actuator system.

When the efficiency is corrected (ignition angle, lambda) within the torque manager, the efficiency is only limited downwards, but no error entry or change to an emergency program, since it cannot be ruled out that the limit value can be undershot in normal operation.

Security queries (error conditions):

- Engine drag torque "md_e_schlepp_hyp" < maximum effective engine torque "md_e_max"
- Torque loss of the motor "md_e_schlepp" > "K_MD_SK_MAX_MDMIN" and Speed threshold "n" > "K_MD_SK_N_MDMIN"
- Output MD dynamic filter > maximum torque "K_MD_SK_MAX"
- Resulting desired moment "md_ind_ Wunsch_red_korr" > "K_MD_SK_MAX"
- Lambda lean factor > 2 (overflow)

Monitoring torque interventions

- Intervention I component of idle control "md_llri" > maximum intervention "K_MD_SK_LLR_MAX"
- Intervention PD component of idle control "md_llrp" > maximum intervention "K_MD_SK_LLR_MAX"

15.2. MONITORING SET TORQUE TO ACTUAL TORQUE

A plausibility check of the actual engine torque for the driver's desired torque over the entire operating range is very difficult, since in this case a large number of input parameters, all transient states and all torque interventions from other modules would also have to be taken into account. This would require almost the entire calculation path to be stored redundantly again, which is not possible due to a lack of resources, or the corresponding tolerance limits would have to be greatly expanded.

Two torque monitoring functions have therefore been implemented in the MSS60. A function that compares the actual torque with the desired torque, taking into account all torque interventions, and has wider tolerance limits. And via torque monitoring, which is limited to a zero torque specification by the driver (PWG = zero), but is activated there accordingly. The advantage of this is that the torque calculation can be estimated much better at this operating point, and the tolerance limit can thus be tightened. Furthermore, it can be assumed that if the engine delivers an undesirably high torque, the driver will automatically take his foot off the accelerator and the activation conditions for this test are therefore met.

15.2.1. MONITORING SET/ACTUAL TORQUE ACROSS THE ENTIRE OPERATING RANGE

Definition of the actual torque md_sk_vergl_ist =

md_ind_ne actually generated indexed actual torque of the engine, determined from Characteristic field over speed and load and ZW efficiency below consideration of all interventions

Definition of the target torque md_sk_vergl_soll =

md_e_fw_filter	filtered driver's desired torque from PWG position or cruise control
- md_e_schlepp +	Drag torque of the motor including all consumers
md_ar + md_llri +	Intervention torque of the anti-judder control
md_llrp	Intervention torque of the I controller of the idling control
	Intervention torque of the P controller of the idle speed control

In the case of a torque-increasing MSR intervention, the maximum of the required torque and "md_sk_vergl_soll" is used as the target torque.

If the actual torque of the motor exceeds the target torque for the period "K_MD_SK_TIMER_MD" by the amount K_MD_SK_OFFSET + (1 - K_MD_SK_WEWICHTUNG) * md_sk_vergl_ist, an error in the Egas system is assumed and a change to emergency program 2 takes place -
Driving via the idle actuator system.

The monitoring is active in the "Engine running" operating state.

15.2.2. MONITORING SET/ACTUAL TORQUE WITH PWG DEFAULT = 0

Activation condition for monitoring

- Operating status Engine is running
- no FGR operation
- no MSR intervention
- Dashpot function of the dynamic filter limited
- Pedal value setting <= K_MD_SK_PWGMIN
- Engine speed > target idle speed + K_MD_SK_NHYS

In this case, if the calculated driver's desired torque exceeds the value "K_MD_SK_FWMAX" or the calculated target DK position exceeds the value "KL_MD_SK_WDK" for the period "K_MD_SK_TIMER", an error in the torque calculation is concluded and the Egas system also switches to the emergency program of the stage 2.

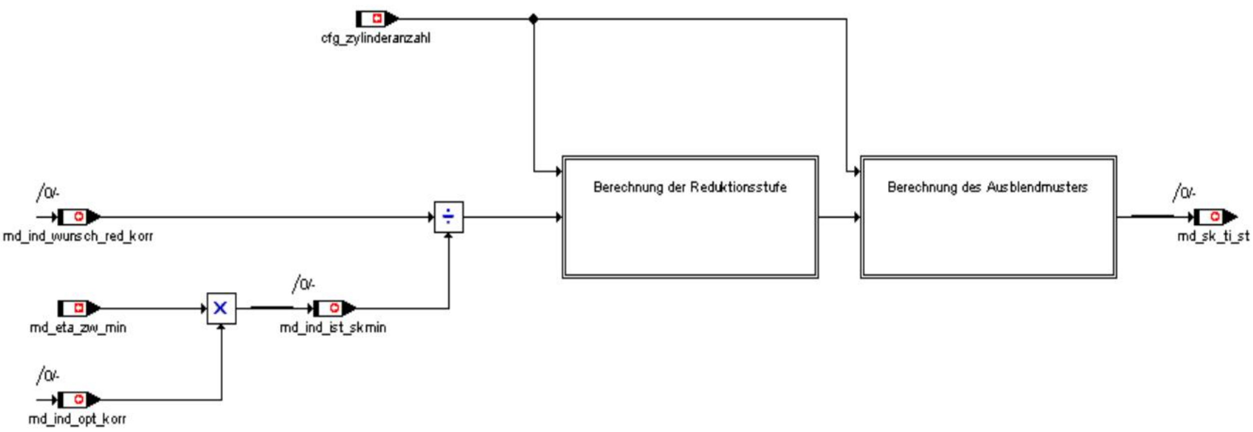
15.3. PARTIAL FIRING WITH OPEN STUCK THROTTLES

If the throttle valves are jammed open, the filling specification calculated by the torque manager can no longer be implemented because the throttle valve system no longer reacts. However, in order to continue to be able to control the torque output of the engine, the torque manager must use another possibility to intervene in torque generation - injection.

For this purpose, based on the desired torque and the current actual torque of the engine, taking into account the possible ignition angle interventions at this operating point, a suppression efficiency "md_sk_verh" is calculated. If this efficiency falls below the value one, this means that the desired engine torque can no longer be reduced solely by intervention in the ignition angle. A suppression level "md_sk_tired" is therefore calculated, which corresponds to the number of active cylinders and switches off the cylinders that are not required according to a predefined suppression pattern.

The intervention in the injection takes place via the variable "md_sk_ti_st", with each cylinder being represented by a bit. If the bit is set, this means that the corresponding cylinder may be active. If the bit is deleted, the cylinder must be switched off.

Overview: partial firing (partial firing.gif)



There are two applicable constants in the partial firing module that are not visible in the overview image. A hysteresis for the efficiency can be set via the constant "K_MD_SK_TIRED_HYS", which acts between fading out and fading in. The constant "K_MD_SK_TIRED_MIN" defines the minimum number of cylinders that must be active. If a reduction level below this value is required, all cylinders are hidden.

16. APPLICABLE DATA OF THE TORQUE MANAGER

Name	importance
K_MD_ASC_BEGR	Minimum torque for ASC intervention
K_MD_ASC_CONTROL	Control byte for activating ASC interventions
K_MD_BEGR_AUSS_ABREG	Curtailment ramp for torque limitation in case of catalyzing dropouts
K_MD_BEGR_AUSS_TIME	Waiting time until torque limitation in the event of damage to the cat dropouts
K_MD_BEGR_DELTA	Change limitation for torque limitation
K_MD_BEGR_RAMPE	Torque limitation ramp
K_MD_BEGR_T	Delay time for torque limitation
K_MD_BEZUG_ZW	Selection for reference torque ZW intervention (control byte ZW interventions)
K_MD_DELTA_SA_HARD	Step size Md filter with hard SA
K_MD_DELTA_SA_SOFT	Step size Md filter with soft SA
K_MD_ETA_EINGRIFF	ZW efficiency below which an ignition angle intervention is only activated
K_MD_ETA_LAMBDA_MAX	maximum lambda efficiency for filling correction
K_MD_ETA_MCS	Configuration parameter for consideration md_eta_zw_stat
K_MD_ETA_STAT_TAU	Filter time constant for station. ZW efficiency
K_MD_I_VMAX	Integrator step size for Vmax control via torque
K_MD_J_FZ	Mass inertia vehicle Nm/s2
K_MD_J_MOTOR	Motor inertia Nm/s2
K_MD_MIN_KKOS_AUS_FILTER	Filter time constant for compressor shutdown
K_MD_MIN_KKOS_FILTER	Filter time constant for compressor activation
K_MD_MIN_KKOS_START	Factor for filter initial value increase compressor activation
K_MD_MIN_START_FILTER	Filter time constant for starting torque regulation
K_MD_MIN_VERH_KRAFTS	Limitation of the MDmin hyperbola with adhesion
K_MD_MIN_VERH_NO_KRAFTS	Limitation of the MDmin hyperbola without adhesion
K_MD_MIN_VERH_START	Limiting MDmin hyperbola during launch
K_MD_MSR_BEGR	Maximum torque for MSR request
K_MD_NORM	Standard torque for CAN interface
K_MD_POLYNOM_A_LL	A polynomial for ignition hook calculation at LL
K_MD_POLYNOM_B_LL	B polynomial for ignition hook calculation at LL
K_MD_RES_CONTROL	Controlbyte Momentenreserve
K_MD_RES_KATH_LL	Md-Reserve at Katheizen im LL
K_MD_RES_KATH_START	Initial value Md reserve for cat heating
K_MD_RES_KATH_T_ABREG	Reduction ramp for torque reserve cat heating
K_MD_RES_KATH_T_AUFREG	Boost ramp for torque reserve cat heating
K_MD_SK_AX_IMIN	Minimum value I controller for acceleration limitation in Egas Notprogramm
K_MD_SK_ETA_MIN	Minimum value for ZW efficiencies

K_MD_SK_FWMAX	max. permissible md_fw_rel at pwg = 0
K_MD_SK_WEIGHT	Weighting factor setpoint to actual torque for torque monitoring
K_MD_SK_LLR_MAX	max. permissible intervention torque of idle speed control
K_MD_SK_MAX	Max. permitted indicated moment within the moment calculation
K_MD_SK_MAX_MDMIN	max. permissible md_ind_min
K_MD_SK_N_MDMIN	Speed threshold for md_min monitoring
K_MD_SK_NHYS	Speed offset to llr_nsoll for monitoring Md zero specification
K_MD_SK_OFFSET	Offset for monitoring setpoint to actual torque
K_MD_SK_PWGMIN	PWG threshold active below Md zero specification
K_MD_SK_TIMER	Filter time for monitoring Md zero specification
K_MD_SK_TIMER_MD	Filter time for monitoring setpoint/actual torque
K_MD_SK_TIRED_HYS	Hysteresis for skip efficiency with partial firing
K_MD_SK_TIRED_MIN	Minimum number of cylinders still active with partial firing
K_MD_STAT_ASC	Test parameters for status feedback DSC intervention
K_MD_TZ_CONTROL	Control byte for the torque manager's ignition angle interventions
K_MD_TZMIN_HYS	ZW hysteresis for SA tripping
K_MD_VMAX_MAX	Minimum torque for I controller Vmax limitation
K_MD_VMAX_MIN	Maximum torque for I controller Vmax limitation
K_V_MAX	V activation threshold for Vmax limitation
K_V_MAX_HYS	V hysteresis for Vmax limitation
KF_MD_DRIVER	Relative target torque from pwg_soll and n
KF_MD_LAMBDA	Lambda actual value with inactive lambda controller = f (n, wi) eg warm-up
KF_MD_MAX_MD_IND_OPT	Motor torque rating = f (n, wi) determined under standard conditions
KF_MD_MIN_BRENN	maximum negative distance to the drag torque of the engine in fired operation = f (n, tmot)
KF_MD_MIN_REIB_DIFF	Friction torque difference to standard temperature = f (tmot, toel)
KF_MD_MIN_START	additional offset torque during start = f(n,tmot)
KF_MD_POLYNOM_A	Parameter f. quadratic term of the ignition hook parabola =f(n,wi)
KF_MD_POLYNOM_B	Parameters for the linear term of the detonator hyperbola = f (n, wi)
KF_MD_RES_KATH	Offset torque for torque reserve cat heating f (n, wi)
KF_MD_RES_KATH_GEW	Weighting factor for torque reserve cat heating f (tmot, t_ml)
KF_MD_WE	Up ramp moment for restart
KF_MD_ZW_OPT	Ignition angle value = f (n, wi)
KL_MD_BEGR_AUSS	Torque limitation = f (n) in the event of cat-damaging misfires
KL_MD_BEGR_FST	Torque limitation = f (n) with an empty tank
KL_MD_BEGR_GANG	gangabh. Maximalmoment = f (gang)
KL_MD_BEGR_NOISE	Torque limitation = f(v) for noise limitation Gear weighting of time
KL_MD_LS_W_GANG	constants for MD dynamic filters
KL_MD_MIN_DN_HYP	Speed offset for MDmin hyperbola = f (tmot)

KL_MD_MIN_REIB_ABREG	Control ramp friction torque offset after start = f (tmot)
KL_MD_MIN_REIB_OFFSET	Reibmoment nach Start = f (tmot)
KL_MD_SK_AX KL_MD_SK_AX_GANG	max. permissible vehicle acceleration in the emergency program
KL_MD_SK_AX_INEG	Gear weighting for acceleration limitation
KL_MD_SK_AX_IPOS	I parameters for acceleration limitation
KL_MD_SK_AX_P KL_MD_SK_GRAD	I parameters for acceleration limitation
KL_MD_SK_MAX KL_MD_SK_WDK	P parameter for acceleration limitation
KL_MD_W_GANG_DASHPOT	Ramp of the transition function at Md limitation
Gangabh. Gewichtungsfaktor f. MD-	max. engine torque in Egas emergency program
Dynamikfilter DASHPOT	Max. permissible DK position with Md zero specification f (n)
KL_MD_WURZEL	Root characteristic for backward calculation of the ignition hook parabola
KL_MD_ZW_LA	Influence of the lambda value on the optimum ignition angle value = f (la)
KL_MD_ZW_TMOT	Influence of the engine temperature on the ignition angle best value = f (tmot)
KL_V_MAX_GANG	gangabhängige Maximalgeschwindigkeit
KL_V_MAX_SK	Maximum speed for Egas emergency program levels

17. TORQUE MANAGER VARIABLES

Name	importance
can_kkos_lm	Load moment air conditioning compressor from CAN
eta_nex x = 1 .. 8	cylinder-selective ZW efficiencies after ZW interventions of the MM
eta_vex x = 1 .. 8	cylinder-selective ZW efficiencies before ZW interventions of the MM
md_ar md_begr_auss	AR intervention
	Limiting torque in the event of cat-damaging misfires or an empty tank
md_begr_auss_st	Status of the torque limitation in the event of cat-damaging misfires or an empty tank
md_begr_st	Torque limit status
md_begr_t	Waiting time until activation of torque limitation
md_dyn_ausg	Output value of the MD dynamic filter
md_dyn_st	Status des MD-Dynamikfilters
md_eta_ausblend	fade-out efficiency
md_eta_kath	Sollwirkungsgrad Katheizen
md_eta_lambda	Lambda efficiency
md_eta_res	Factor for increasing the filling specification with torque reserve
md_eta_zw_eingr	ZW target efficiency before intervention
md_eta_zw_min	ZW efficiency for the latest possible ignition angle
md_eta_zw_ne	Average ZW efficiency across all cylinders after torque intervention
md_eta_zw_soll	Target efficiency before intervention
md_eta_zw_stat	ZW efficiency for stationary ignition angle
md_eta_zw_ve	Average ZW efficiency across all cylinders before torque intervention
md_e_fw md_fw_filter	Desired moment driver/FGR effective
md_fw_rel md_ind_asc	filtered desired moment driver/FGR
md_ind_asc_abs	relative driver desired torque
md_ind_asc_lm	Momenteneingriff ASC
md_ind_asc_lm_abs	indicated torque for ASC ignition angle intervention
md_ind_fgr md_ind_ist	Torque intervention ASC via filling
md_ind_ist_skmin	indicated moment for ASC filling procedure
md_ind_lm_ist	Desired moment from FGR
md_ind_max md_ind_msr	Driver desired torque without interventions / corrections
	minimum achievable actual torque at tz_min
	Desired torque with torque limitations
	maximum indicated moment
	Momenteneingriff MSR
md_ind_msr_abs	indicated moment for MSR filling procedure
md_ind_ne	Actual torque including ignition angle interventions by the torque manager
md_ind_ne_ist	determined indicated torque including all interventions
md_ind_opt	maximum actual torque under standard conditions

md_ind_opt_korr	maximum actual torque under the current conditions
md_e_schlepp	Effective drag torque of the motor
md_ind_ve	Actual torque without ignition angle interventions by the torque manager
md_e_verbraucher	Torque loss effectively by consumers
md_ind_vmax	maximum torque at Vmax limitation
md_ind_wish	indicated desired torque from the driver / FGR
md_ind_wish_begr	limited moment of desire
md_ind_wish_filter	=md_ind_wish + md_llri (doesn't exist in Gredi but SW)
md_ind_wish_korr	Desired moment for filling path after 1st correction level
md_ind_wish_red	Desired moment for filling path after 2nd stage of correction
md_ind_wish_red_korr	Desired moment for filling path after 3rd correction stage
md_kr_dtz_mittel	mean ZW late drawing per cylinder from KR/KA
md_ksg md_ksg_filter	KSG Istmoment
md_llr_tz md_llra	KSG Istmoment gefiltert
	TZ share of the LFR
	LFR health adaptations
md_llra_ko	Adaptation share of the LFR at B_KO
md_llri md_llrp	I part of the LFR
md_ls_kf	P component of the LFC
md_max_begr	Input value for LS filter
md_mcs_zyl	maximum permitted indicated engine torque
md_min_dn_hyp	Calculation of ignition angle intervention for cylinder x active
md_min_start	Speed offset for calculating MDmin hyperbola
md_noise_max	Offset torque for start
	Limitation of the indicated moment to reduce the noise (not in the Gredi)
md_norm	Reference torque for CAN interface
md_norm_can	Normalization reference for torque interface
md_polynom_a	current parameter for quadr. Term of the ignition hook parabola
md_polynom_b	current parameter for linear term of ignition hook parabola
md_reib	determined friction torque
md_reib_abreg	Incremental regulation of frictional torque
md_reib_offset	Offset Reibmoment
md_res md_res_kath	currently effective torque reserve
md_res_kath_faktor	Torque reserve for cat heating
md_res_kath_roh	Ramp-down factor torque reserve cat heating
md_sawe_filter	Raw torque reserve for cat heating
md_sawe_verh	filtered desired moment SAWE
md_sk_begr md_sk_ti_st	Start time of the load impact filter
md_sk_tired	Maximalmoment MD-SK
	Release mask for injection channels
	Number of enabled injection ports

md_sk_vergl_ist	Actual torque in MD monitoring
md_sk_vergl_soll	Target torque in MD monitoring
md_sk_verh md_st	Ratio of target torque to min. actual torque
md_st_eingriff	Statusbyte Momentenmanager
md_st_tz	Status Momenteneingriff
	Statusbyte Zuendwinkeleingriff Momentenmanager
md_tz_red	Torque specification tz intervention
md_wish_rel	relative desired torque driver/FGR
md_zw_lambda	Lambda compensation optimal ignition angle not in Gredi
md_zw_opt	Optimum ignition angle
md_zw_opt_korr	Ignition angle best value under the current conditions
md_motor	engine torque delivered to the clutch including all interventions =md_ind_ne - md_e_schlepp
wi	specific indicated work [kJ/dm3]

Project: MMSS54

Module: Antiruckelfunktion

Table of Contents

change documentation

Version: 1.0 02.11.2004
initial creation

8. Antiruckelfunktion AR

8.1 General

In the case of a rapid transition from overrun or low partial load to higher load ranges, vibrations can occur in the drive train in the lower speed range. The anti-judder function of the MSS54 counteracts these bucking vibrations by detecting the vibrations of the drive train and damping them through in-phase torque interventions.

During a positive speed gradient (increasing engine speed), the engine torque delivered is reduced by means of a torque intervention (ignition angle retard).

8.2 Calculation of speed gradients for AR

The calculation of the speed gradient "d_n_segment" is based on the segment speed "n_segment" and is repeated every 120° crank angle. The segment speed is calculated from the duration of a segment (60° CA before TDC to 60° CA after TDC).

Calculation formula:

d_n_segment(t)= ((n_segment(t) - n_segment(t-120°) / t_segment(t) + d_n_segment(t-120°)) / 2

speed gradient = Mean value from the speed difference of two segments, normalized rpm/sec and the previous gradient

8.3 AR Activation Condition

To activate the anti-judder function, the following conditions must be met:

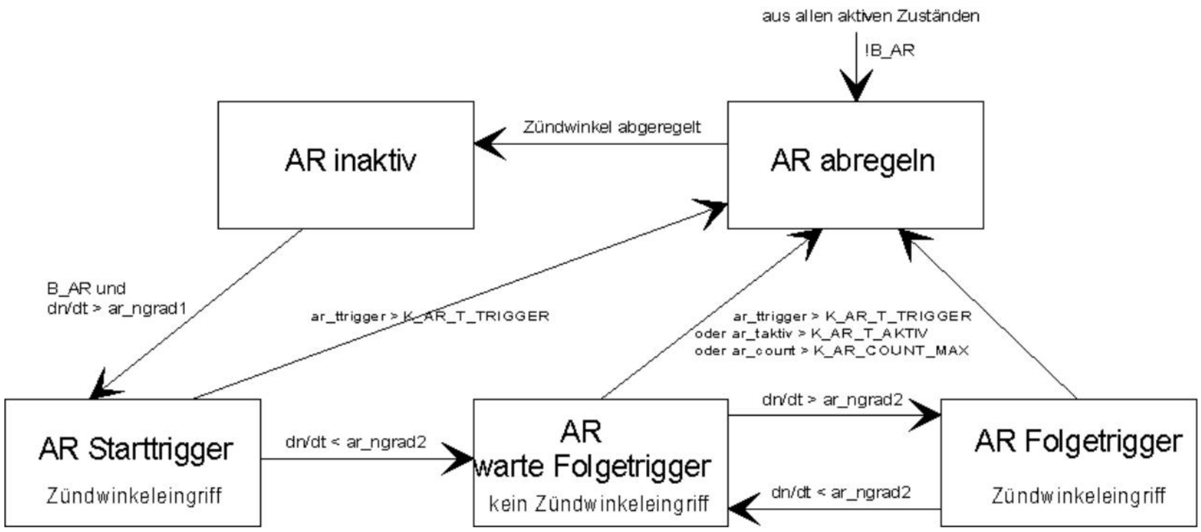
```
B_AR =        B_TL                                   ; Part load operating state
     and tmot ÿ K_AR_TMOT_MIN and                    ; Motor temperature greater than threshold
     K_AR_NMIN ÿ n ÿ K_AR_NMAX and                   ; RPM within range
     K_AR_RFMIN ÿ tl ÿ K_AR_RFMAX ; load within range
     and      ( K_AR_VMIN ÿ v ÿ K_AR_VMAX or         ; speed within range
                B_V_FEHLER ) and S_GANG              ; as long as V detection is error-free
                                                     ; adhesion present
                                                     ; (not active at the moment)
```

The activation condition is summarized for the following documentation to condition B_AR.

B_AR = 1 conditions met
B_AR = 0 condition not met

8.4 States of AR

Figure 8.1: State machine of the anti-judder function



As long as the condition B_AR is not met and the ignition angle interventions of previous AR regulations are limited, the AR is in the "inactive" state.

If the B_AR condition is met, the AR changes to the "start trigger" state as soon as a speed gradient greater than the trigger threshold ar_ngrad1 (from characteristic map KF_AR_NGRAD1 = f(n, tl)) is detected.

If the speed gradient falls back below the value ar_ngrad2 (from map KF_AR_NGRAD2 = f(nt, tl)), the AR changes to the "Wait for subsequent trigger" state. If this gradient threshold is not undershot within the time K_AR_T_TRIGGER after recognizing the start trigger, it is concluded that there is no bucking and the AR changes to the "slow down" state.

In the "Wait for subsequent triggering" state, no AR ignition angle intervention is active. As soon as the speed gradient exceeds the threshold ar_ngrad2 again, the AR changes to the "sequential triggering" state. If this threshold has not yet been reached, the AR switches to the "curtail down" state as soon as one of the following conditions is detected.

• Time since last state transition > K_AR_T_TRIGGER
• Total time of AR active • > K_AR_T_AKTIV
Number of triggers > K_AR_COUNT_MAX

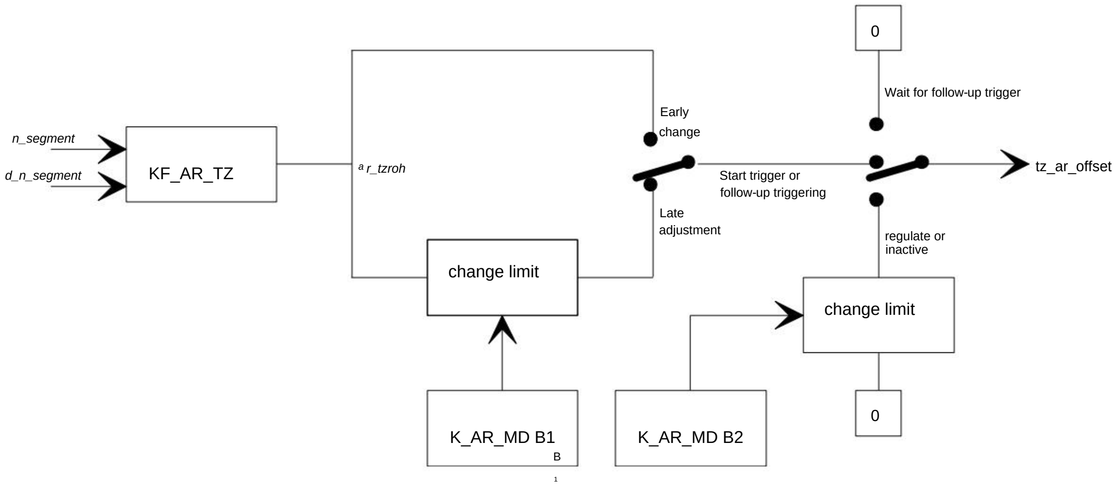
In the "sequential triggering" state, an ignition angle intervention is again active. A change to the "Wait for subsequent triggering" state occurs when the speed gradient becomes smaller again ar_ngrad2 and the conditions for ending the AR are not yet met.

The AR should be ended in the "regulate" state. Any existing ignition angle interventions are limited. Then there is a change to the "inactive" state.

The following applies to all active statuses: As soon as the B_AR condition is no longer met, there is a change to the "regulate" status.

8.5 AR ignition angle intervention

Figure 8.2: Calculation of the ignition angle offset



8.6 Data of AR

Variable der AR:

Name	Meaning of	type resolution	
ar_mdroh	unfiltered torque offset of the AR	sw 1/10 ° KW	
ar_grad1	Gradient threshold for AR triggering sw 1 rpm/s		
ar_grad2	Gradient threshold for AR follow-up triggering sw 1 rpm/s		
ar_taktiv	System time at which AR was last activated uw 1 ms		
ar_ttrigger	System time at the last AR trigger uw 1 ms		
ar_count	Number of triggers uc 1		
ar_zustand	Status variable of the AR (only 1 bit set each) Bit 0: AR inaktiv 1: Start trigger detected 2: wait for subsequent trigger 3: Consecutive trigger detected 4: Limit AR interventions	etc.	
ar_md_offset	AR ignition angle offset	sw 1/10 ° KW	

Application data of the AR:

Name	importance
K_AR_TMOT_MIN	lower temperature threshold for AR
K_AR_NMIN	lower speed threshold for AR
K_AR_NMAX	upper speed threshold for AR
K_AR_RFMIN	lower filling threshold for AR
K_AR_RFMAX	upper filling threshold for AR
K_AR_VMIN	lower speed threshold for AR
K_AR_VMAX	upper speed threshold for AR
K_AR_T_TRIGGER	max. period for the next triggering
K_AR_T_AKTIV	maximum AR active time
K_AR_ANZ_TRIGGER max. Number of triggers	
K_AR_MDB1 Ignition angle change limitation for AR retards	
K_AR_MDB2 Ignition angle change limitation for AR curtailment (early)	
KF_AR_NGRAD1 gradient threshold for start triggering = f(n,rf)	
KF_AR_NGRAD2 gradient threshold for subsequent triggering = f(n,rf)	
KF_AR_MD Ignition angle offset of AR = f (n, dn / dt)	

PROJECT: MSS54

MODULE : OPERATING MODE MANAGER

AUTHORISATION

AUTHOR (ZS-M-57)__________________________ DATE __

APPROVED (ZS-M-57) ____________________ DATE ___________

APPROVED (EA-E-2)_____________________ DATE ___________

	Department	date	Name	filename
Author	ZS-M-57	02.08.04	Frank	1.02

Changes:

Version	date	comment
S310	2.8.2004	First version
S320	8.11.2004	Minihub mode added
S330	1.12.2004	Calculation of wi moved to the torque manager
S330	2004-12-01	Renamed evt_state to bm_evt_state
S330	12/4/2004	Minihub changed from 4V to 3V
S340	12/8/2004	Manual mode no longer has its own operating mode (bm_evt_state)
S360	February 20, 2005	Changed the picture of the hysteresis of KF_BM_AUSWAHL, it was misleading
S360	5/30/2005	Operating mode transitions now implemented
S370	1.7.2005	4-stroke braking mode added
S380	10/18/2005	12-stroke operating mode newly added
S380	10/18/2005	Operating mode transitions in the documentation revised
S380	11/2/2005	When transitioning from braking operation AÖ=140KW previously AÖ=180KW
S380	11/2/2005	Transition from ZAS changed: AÖ now implemented via KL

Table of Contents

1 FUNCTIONAL DESCRIPTION

In order to enable optimal throttle-free operation of the EVT engine in the entire operating range, various operating modes of the valve train must be set. For this purpose, a suitable operating mode is selected in this function depending on the load and speed. The operating modes currently used are briefly described below:

1.1 DESCRIPTION OF THE OPERATING MODES

The following designations are used for the following description:

designation	description
UTH	Bottom dead center before the high-pressure phase (combustion)
TO THE	outlet opens
AS	Outlet Closes
IT'S THE	Inlet opens
IS	Inlet Closes
Cycle, work cycle	The cycle or work cycle refers to the entire engine process that begins with the gas exchange when the exhaust valve is open. The first valve activities are thus EO and AS. The gas exchange is then terminated by ES. Compression and combustion with expansion now take place. The last action of a working cycle is AO.
Cycle consistency	Cycle consistency means that all valve control times EO, AS, ES and AO, as well as ignition and injection, are kept together for each working cycle in each cylinder. The cycle consistency only has an effect on dynamic processes. Cycle consistency is an important prerequisite for an EVT engine control unit, since due to the digital control of the valves from cycle to cycle, each valve timing can be changed and it must be ensured that all parameters of a cycle match.
smes	Indicated mean pressure [bar]. Calculation: Integral p dV over one cycle divided by cylinder volume
wi	Indicated specific work [kJ/dm3]. Calculation: Integral p dV over one cycle divided by cylinder volume (corresponds to the value of pmi * 0.1)

1.1.1 CLOSING TIME OF INTAKE VALVES

These methods of load control differ in the position of the closing time of the intake valve. Both FES and SES can be combined with all other processes, eg cylinder deactivation, mini-stroke or the 12-stroke process.

1.1.1.1 FES (Early Admission Closes)

In the FES operating mode, the intake valve is closed before the UTH to set a desired torque. After closing the intake valve, expansion takes place up to UTH.
Since the subsequent compression in the pV diagram is almost on this expansion line, this does not result in any losses.

1.1.1.2 Late Entry Closes (SES)

Although the SES operating mode has disadvantages in terms of consumption and dynamics compared to FES, it is used at higher engine speeds (above 4000 rpm) because the positioning speed of the actuators is not sufficient at high speeds to implement the FES operating mode. In the SES operating mode, the intake valve is closed after the UTH to set a desired torque. At low torques, the point in time at which the intake valves close would be so late that the intake manifold would overheat and the point in time at which the intake valves would close would be close to the ignition point. Therefore, the load range for SES is limited to a load of pmi = approx. 5 bar. To set lower engine loads, the SES process must be combined with cylinder deactivation or an i-stroke process (eg 12-stroke process). This combination increases the load per cylinder fired again.

1.1.2 NUMBER OF OPERATING VALVES

These operating modes differ in the number of valves actuated per working cycle. They can be combined with all other processes, eg cylinder deactivation or the i-cycle process.

1.1.2.1 4 valve operation (4V)

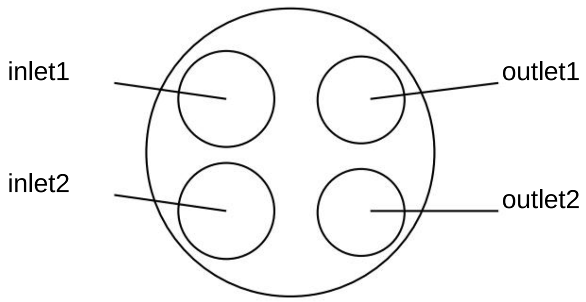
In 4V operation, 2 intake valves and 2 exhaust valves are actuated per working cycle.

1.1.2.2 3 valve operation (3V)

In 3V operation, 2 inlet valves and 1 outlet valve are actuated per working cycle. In order to achieve an even load on both exhaust valves, the other exhaust valve is actuated alternately in each working cycle.

1.1.2.3 2 valve operation (2-V)

In 2V operation, 1 inlet valve and 1 outlet valve are actuated per working cycle. Since only one injection nozzle is used per cylinder, which injects the fuel into both intake ports, both intake valves are actuated alternately from work cycle to work cycle. The outlet valve, which is arranged diagonally, is actuated in each working cycle. This means that, for example, inlet 1 and outlet 2 are actuated in one working cycle and inlet 2 and outlet 1 in the following working cycle (see figure). Due to the symmetrical arrangement of the channels, a reproducible gas exchange is achieved.



1.1.3 CYLINDER DEACTIVATION

Cylinder deactivation deactivates cylinders 2 and 3, ie only cylinders 1 and 4 are fired. The valves of the deactivated cylinders are kept in the closed state.

1.1.4 12- STROKE OPERATION

The 12-clock method corresponds to a 4-clock method in which 8 empty bars are inserted. One stroke corresponds to 180 degrees of crank angle in a 4-cylinder, i.e. a complete upward movement or a complete downward movement of the piston. Thus, a working cycle of a 12-
cycle process, i.e. 6 crankshaft revolutions.

In the 12-stroke process operating mode (bm_evt_state = 7), all cylinders are fired once during the period in which the first cylinder is doing 3 work cycles. This changes the distance between the high-pressure processes (see Table 1).

		4 –Takt	12 - Takt
N44 / 4 – cylinder	FOR	180 ° KW	540 ° KW
N64 / 8 - cylinder	FOR	90 ° KW	270 ° KW

Table 1: Distance of HD processes N44 / N64

By increasing the number of cycles, the operating point shift and also the operating range with the load control method SES is further expanded. One can speak of an increased form of cylinder deactivation. The 12-cycle process takes place in 4V operation.

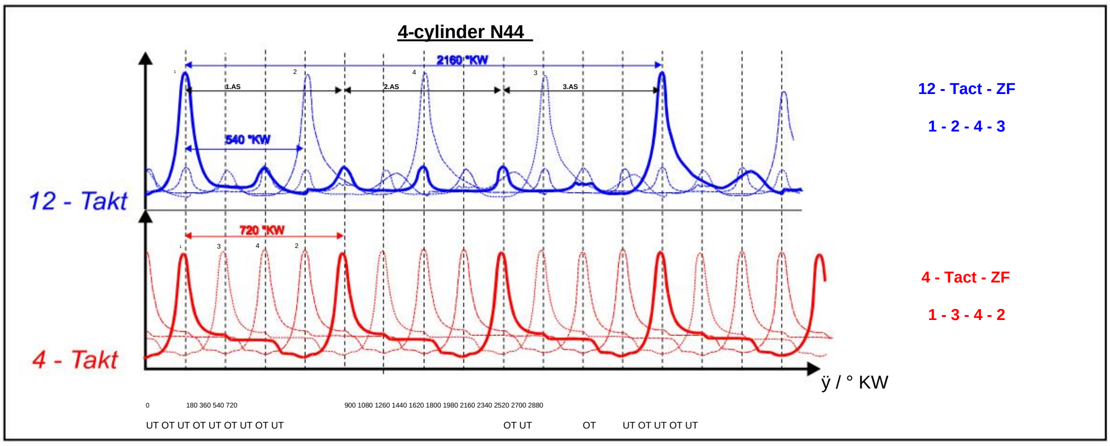


Figure 1.1: Pressure curves in the 12-stroke process (4-cylinder, comparison to 4-stroke Operation)

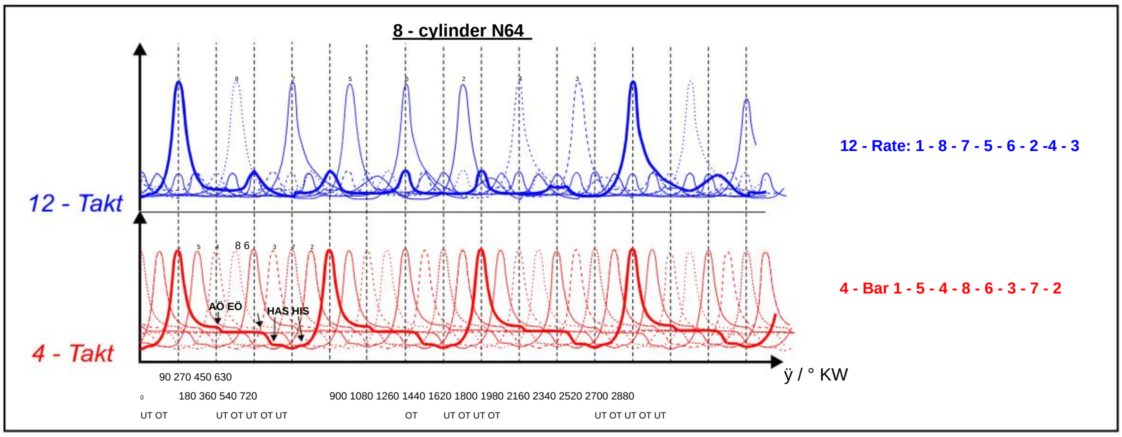


Figure 1.2: Pressure curves in the 12-stroke process (8-cylinder, compared to the 4-stroke Operation)

Advantages of 12-stroke operation:

- Wall film effects in transient operation due to the same ignition intervals for all cylinders reduced
- Cooling down of the cylinders is avoided by alternating "switching off" of the cylinders
- Reduction in consumption by shifting the operating point
- Reduction in valve train performance
- Idle cycles can be displayed with almost no gas exchange losses

The control edges for EÖ and ES are activated in the 12-stroke process in the 3rd ASP, the control edges for AO and AS in the 1st ASP of the respective cylinder. This results in a high-pressure process in every third ASP.

Rate	1	2	3 4	5 6 7	8 9					10	11	12
occurrence cylinder	Pushing out expansion										suction	Compre ssion
occurrence valves	AÖ	AS	valves closed							EÖ	IS	
	ÿ		~~A injection~~							A injection		

Table 2: Course of the cycles in 12-cycle operation (no time representation)

1.1.5 MINIHUB FOR INLET

In contrast to the normal full lift process, the mini lift process keeps the valves at a small valve lift (mini lift). This is made possible by controlled actuator operation using an actuator stroke sensor. The intake manifold pressure waves are reduced by the mini-stroke because the valves are not closed so abruptly. The valve touchdown speed can be minimized more easily because only small strokes and speeds are used.

In addition, for low engine speeds and engine loads, a possibility of generating turbulence and mixture preparation is offered, which can improve engine efficiency. The mini stroke can also be changed as a parameter. The size of the mini-stroke is therefore available as an additional parameter for setting a motor load. In the mini lift operating mode, only the inlet is operated with a small amplitude and the outlet with a full lift. Two inlet valves and only one outlet valve (3V) are operated in a toggled manner.

1.1.6 BRAKING OPERATION 4- STROKE

The motor can be used for braking thanks to special valve control times, with the braking torque being infinitely adjustable via the valve control times. Only the outlet valves are used to prevent the fresh mixture from being flushed through. In the area of bottom dead center, the exhaust valves are closed. Compression then takes place until a desired pressure is reached, at which point the outlet valve is opened. The compressed gas now flows out of the cylinder into the exhaust system. As the piston moves down, gas from the exhaust system is sucked into the cylinder. To achieve maximum braking effect, the procedure described should be repeated for each crankshaft revolution. This corresponds to a 2-cycle process.

Braking should only be possible from a speed of twice the no-load speed (currently n=1400 rpm). During braking operation, a high negative **pmi** is intentionally generated to decelerate the vehicle without the mechanical brakes.

When braking, no fuel is injected and gas exchange losses are only realized via the exhaust valves. Negative torque is achieved by closing the exhaust valve around bottom dead center and opening at a certain cylinder pressure, thus generating compression losses. The entrances remain closed.

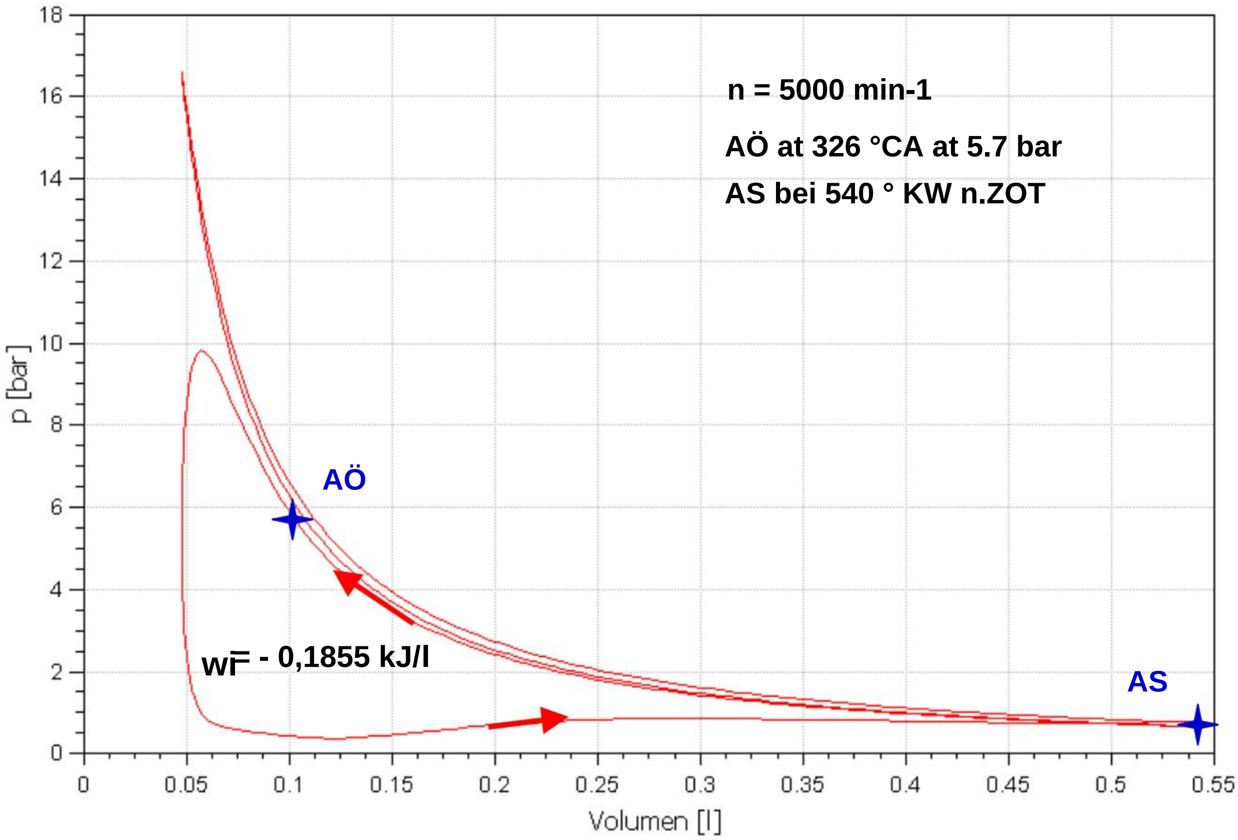


Figure 2 4-stroke brakes

With four-stroke braking, the exhaust valve is closed around bottom dead center (540° crank angle after ZOT). The position of AS thus determines the respective filling. Depending on the desired braking torque, AÖ is between UT and TDC (180°….360°CA after ZOT). The later the opening time is, the higher the compression losses and thus the braking performance to be achieved.

The maximum possible braking torque is limited by the cylinder pressure at the point in time AÖ. If the cylinder pressure at AO is too high, the valve cannot open against the gas force and will only open at an unspecified point in time after TDC, which means that the braking performance cannot be clearly defined.
As a result, the braking torque would also be lower, since the valve opens in the decompression phase.

Braking is only implemented via the exhaust valves to prevent air from being pushed through. In addition to the acoustics, the high pressure amplitudes would also be problematic if the braking operation were implemented on the intake side.

1.2 TRANSITIONS OF OPERATING MODES

In contrast to the transition functions in conventional engines, these transitions only consider the transition from one work cycle to the next, specifically for each cylinder individually. Initially, only the valve train-specific transitions should be implemented. The fuel path transitions will be implemented at a later date. In the following description, the first working cycle is referred to as working cycle_1 and the subsequent one of a cylinder is referred to as working cycle_2.

1.2.1 DIFFERENT NUMBER OF ACTIVE EXHAUST VALVES

Since the opening of the exhaust valves (AO) is always the last action in a working cycle, the number of open exhaust valves does not match the number of valves to be closed when changing from 4V operation to 3V or 2V operation. In this case, AÖ belongs to work cycle_1 and AS to work cycle_2, ie 2 exhaust valves are opened, but only one exhaust valve is closed. In this case, a special treatment must be applied that closes the 2nd exhaust valve at the same time as the first exhaust valve. In the reverse transition from 3V or 2V operation to 4V operation, only one exhaust valve is opened, but both exhaust valves are intended to be closed. In this case, the closing of the 2nd exhaust valve must be suppressed.

1.2.2 TRANSITION TO CYLINDER DEACTIVATION

The transition from a working cycle in which a cylinder is operated to a working cycle in which the cylinder is switched off should take place as follows:

1. normal outlet opening as the last action of the fired working cycle
2. Outlet closing at gas exchange TDC

All valves are now closed and as long as cylinder deactivation is active, no valves should be actuated.

1.2.3 TRANSITION FROM CYLINDER DEACTIVATION

In the PV diagram of a deactivated cylinder, the compression and expansion lines are almost congruent. A strong negative pressure is achieved at bottom dead center.
If you were to open the exhaust valve at this point in time, the exhaust gas would come out of the exhaust system flow into the cylinder at the speed of sound, stirring up oil. During the subsequent compression, this oil would enter the exhaust system unburned. To avoid this problem, the outlet valve should be opened as late as possible.

The transition from a working cycle in which one cylinder is switched off to a fired one
Working cycle should take place as follows:

1. Outlet opening is replaced by the value from the **KL_BART_AO_ZAS** characteristic curve .
2. Now all valve control parameters of the fired cycle can be used. (Don't forget that all open exhaust valves must be closed!)

1.2.4 TRANSITION FROM FIRED TO BRAKING OPERATION (4-STROKE)

During the transition from fired operation to braking operation, the AÖ control edge from fired operation is still used for the first ASP and the AS control edge is already used from braking.
The inlet control edges are suppressed.

Example (fired operation -> braking):

crossing

fired	crossing	brakes	brakes
AÖ FES = 180	AÖ FES = 180	AÖ BB = 320	AÖ BB = 320
AS FES = 360	AS BB = 540	AS BB = 540	
EÖ FES = 360	EÖ BB = --	EÖ BB = --	
ES FES = 440	ENBB_ = --	ENBB_ = --	

fired → brakes

1.2.5 TRANSITION FROM BRAKES TO FIRED (4-STROKE)

When changing from braking to fired operation, the already calculated control edge for AÖ (with the value AÖ=140°CA after TDC) must be overwritten.

Example (brakes -> fired operation):

crossing

brakes	crossing	fired	fired
AÖ BB = 320	AÖ BB = ~~320~~ 140	AÖ FES = 180	AÖ FES = 180
AS BB = 540	AS FES = 360	AS FES = 360	
EÖ BB = --	EÖ FES = 360	EÖ FES = 360	
ENBB_ = --	ES FES = 440	ES FES = 440	

brakes → fired

1.2.6 TRANSITION FROM ZAS TO 12-STROKE

When changing from operation with ZAS, the control edges for AÖ and AS are activated in the first ASP, the control edges EÖ and ES are not activated.
The control edges AÖ and AS are only activated again 3 ASP later, the control edges for EÖ and ES are activated again after 8 idle cycles.

The injection for all cylinders must be activated and, as shown in Table 2, must be controlled with 8 idle strokes.

It must be ensured that the transition between the two operating modes is only possible at the point in time when there is an overlap in the high-pressure process of the active cylinders of both operating modes. (ÿ see Table 3 and Figure 1.4.: Changing from ZAS to 12-stroke is only possible in the areas marked in red.)

	ZAS ÿ 12 T	12 T ÿ RANGE	4 T ÿ 12 T	12 T ÿ 4 T
N44	1080	5401	1802	540
N64	540	2701	902	270

Table 3: KW difference for possible BA change

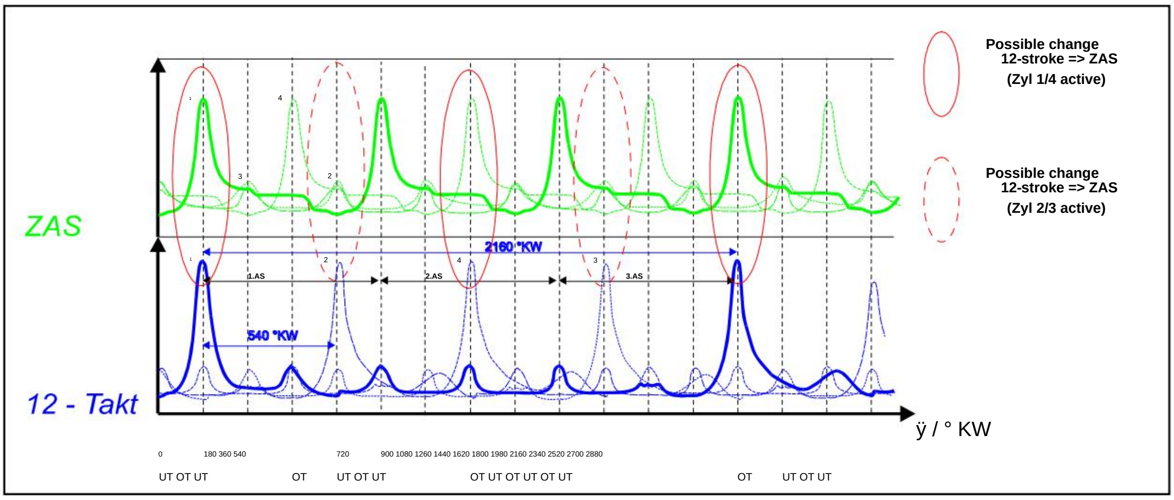


Figure 3: **Possible transitions from ZAS to 12-stroke (with ZAS Zyl 1und 4 active)**

---

1 Change to the next possible cylinder group
2 change in stages

1.2.7 TRANSITION FROM 12- STROKE TO ZAS

When changing from 12-stroke operation to ZAS operation, the control edges for the inlet valves must be activated again in the 1st ASP.
The cylinders are activated according to the firing sequence 1-3-4-2 (4-cyl), cylinders 2 and 3 (or 1 and 4) are not activated (=> ZAS operation).
A change from 12-stroke to ZAS is possible every 540 °KW [270 °KW]. Depending on which cylinder is overlapping in the HP process at the time of the change, a decision is made as to which group changes to ZAS operation.
The 12-bar ÿ ZAS transition does not have to occur at the times when there is an overlap in the HD process.

1.2.8 TRANSITION FROM 4-STROKE TO 12-STROKE

The change from 4-stroke to 12-stroke operation can take place every 180 °CA [90°CA] (ÿ Table 3).
The change takes place "in stages", ie each cylinder that has completed the HP process in 4-stroke changes to 12-stroke operation and is then operated with a new firing order and firing interval.

1.2.9 TRANSITION FROM 12- STROKE TO 4-STROKE

The change from 12-clock to the other operating modes takes place in the same way as the transition from 12-clock ÿ ZAS. However, all cylinders are controlled according to the "normal" firing order.
The transition from 12-stroke operation to 4-stroke operation can take place every 540 °CA for 4-cylinder engines, but generally always at the point in time when the HP processes overlap. (ÿ see Figure 4 A change from 12-bar to 4-bar is only possible in the magenta-marked areas.)

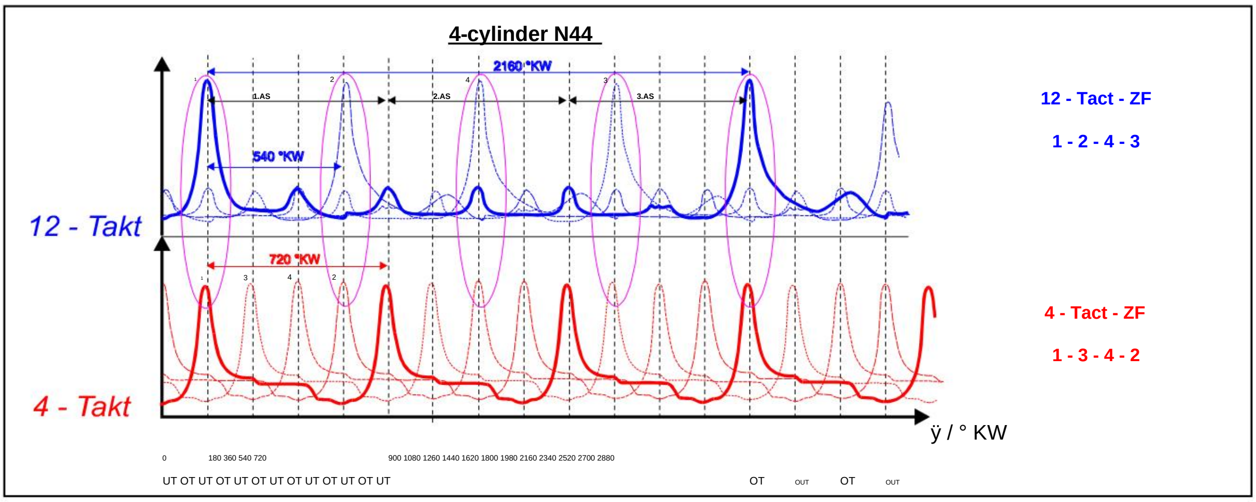


Figure 4: **Possible transitions 12-bar ÿ 4-bar**

1.3 DUTY CALCULATION

The main part of the function is formed by the look-up **tables KF_BM_AUSWAHL** and **KF_BM_AUSWAHL_KATH** for catalytic converter heating (only in the next SW version!!!). This look-up table calculates the operating mode **bm_evt_state** via the inputs **wi** and **n** without interpolating the z values.
Table 4 shows the definition of **bm_evt_state:**

bm_evt_state 0 1	**operating mode**
2 3 4 5 6 7 8 9 10	Cylinder deactivation + SES + 4 valves
11 12 13	Cylinder deactivation + FES + 3 valves
	FES / 2V
	FES / 3V
	FES / 4V
	SES / 4V
	Brake 4 Takt
	12 Clock / 4V
	Katheizen / 3V
	Katheizen / Minihub
	Cat heating / cylinder deactivation
	Minihub / 3V
	Vollast / 4V
	Start

Table 4 Operating modes

In order to prevent jumping back and forth between the states in a load state, the inputs are switched via a hysteresis.

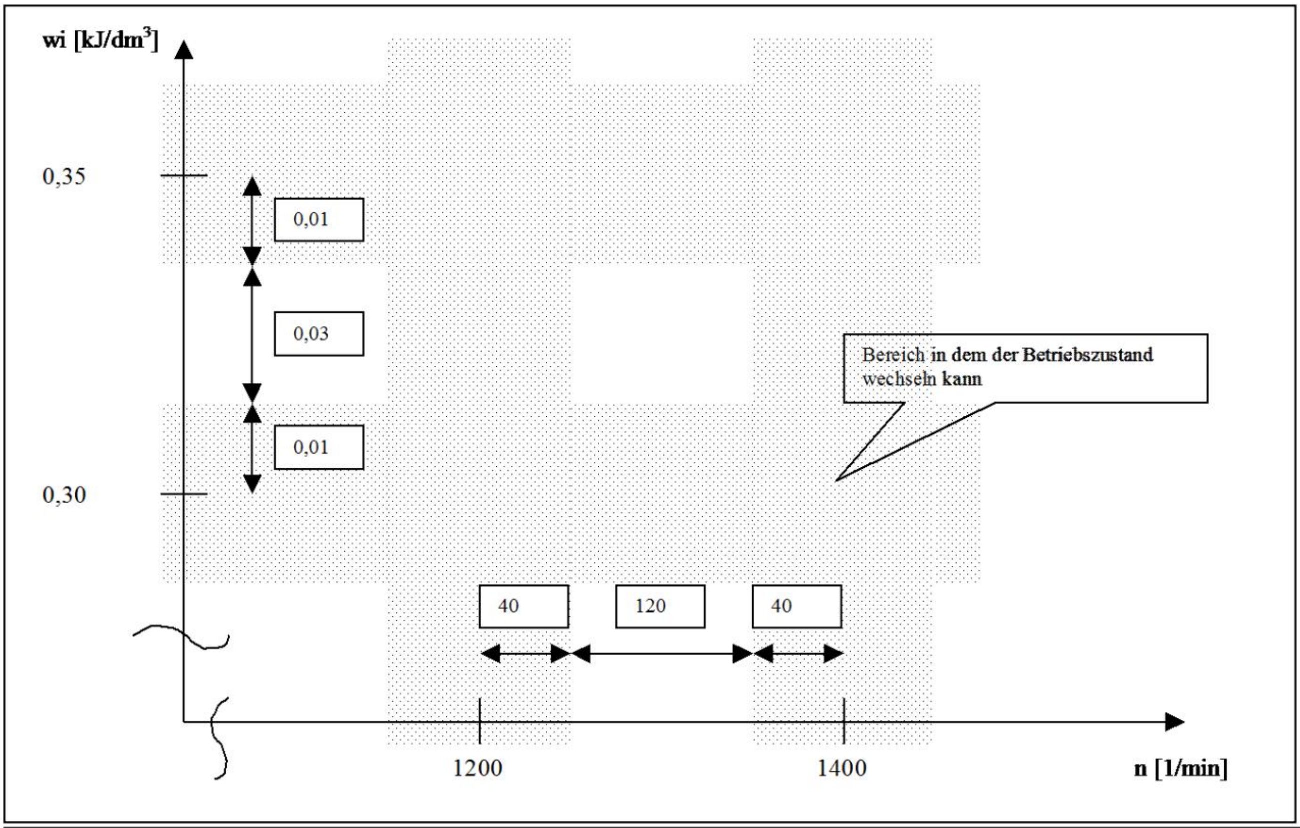


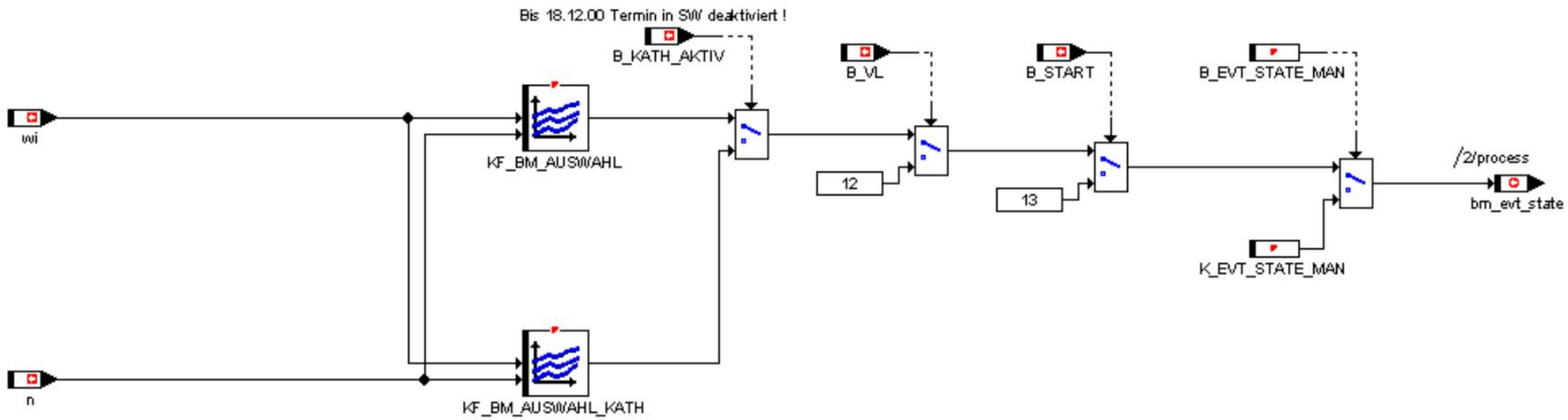
Figure 5 Hysteresis of the axes of KF_BM_AUSWAHL

The state can only change when the input variable enters the shaded areas. The spaces remain undefined and **bm_evt_state** keeps the last value. The distance between the support points of the speed axis must not be less than 200 rpm. The same applies to the wi axis, here too the distance must not be less than 0.05!

When defining the support points for the various operating modes in **KF_BM_AUSWAHL** and **KF_BM_AUSWAHL_KATH ,** it is important that they correspond to the limit support points of the associated basic data set - control edges, ignition angle, air mass, throttle valve angle and advance angle (see evt_momentenrealisierungs.doc)!

Machine Translated by Google

1.4 FUNCTIONAL DIAGRAM



2 OPERATING MODE MANAGER DATA

The function is calculated in the angle-synchronous task in the master.

Description of the variables:

bm_evt_state	operating status	where

Description of the application data:

KF_BM_SELECTION	Operating state map, possibly	your/your/ub
KF_BM_AUSWAHL_KATH	Map operating status possibly with cat heating	your/your/ub
B_EVT_STATE_MAN	Switch to manual specification of bm_evt_state manual bm_evt_state	where
K_EVT_STATE_MAN	specification	where
KL_BART_AO_ZAS	Control edge AO when transitioning from ZAS	your/your

PROJECT: MSS54

MODULE: EVT TORQUE REALIZATION

AUTHORISATION

AUTHOR (ZS-M-57)__________________________ DATE ___________

APPROVED (ZS-M-57) ____________________ DATE ___________

APPROVED (EA-E-2) _____________________ DATE ___________

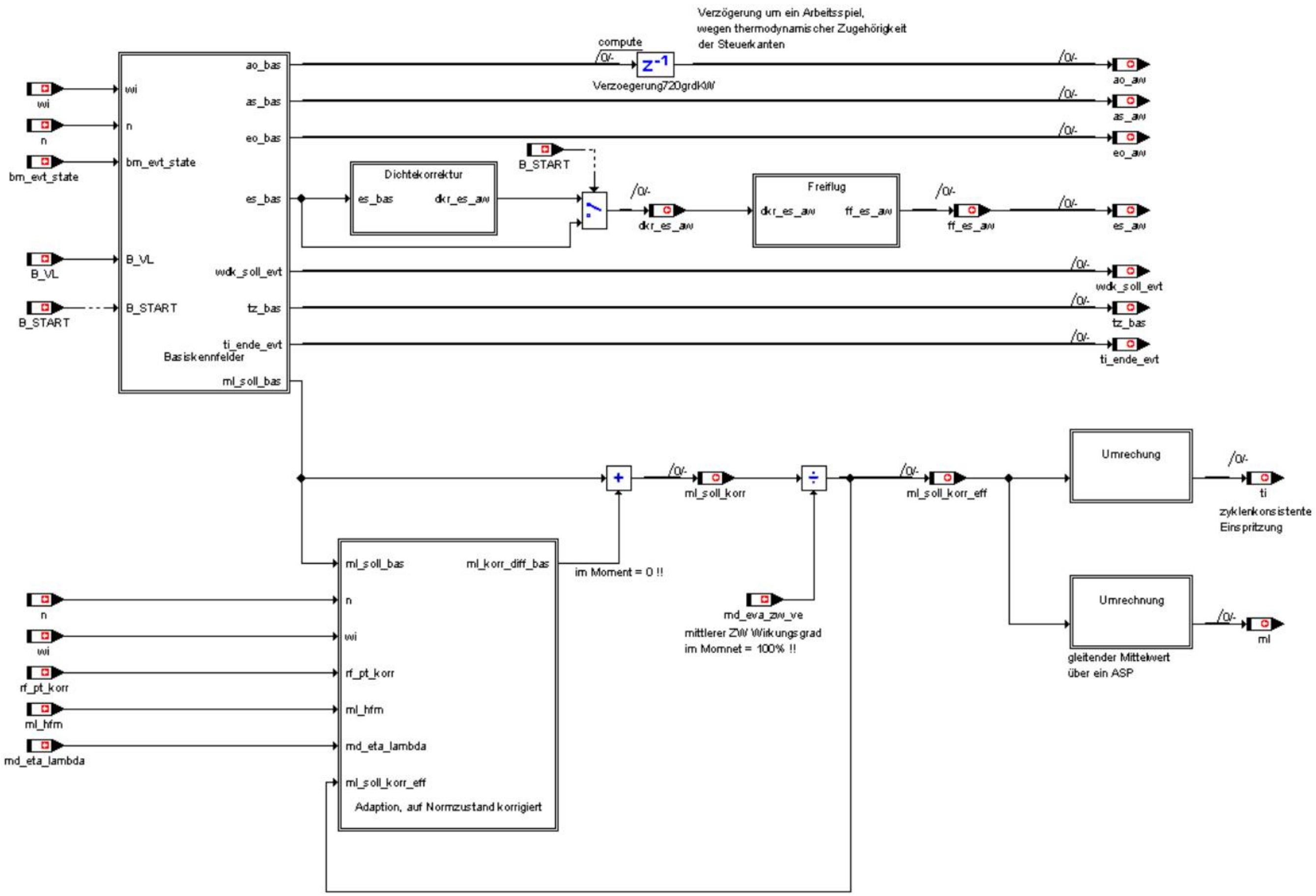
Changes:

Version	date	comment
r310	08/31/2004	First version
r320	10/27/2004	Minihub added
r320	11/06/2004	Conversion of the air mass to [mg/l*ASP]
r320	11/06/2004	Forward angle refers to ES
r330	12/04/2004	Minihub changed from 4V to 3V
r370	27.03.2005	4-stroke braking operation added
r390	April 25th, 2005	ti_ende and es control edges at the start of K->KF extended Calculation of the density correction in the start changed

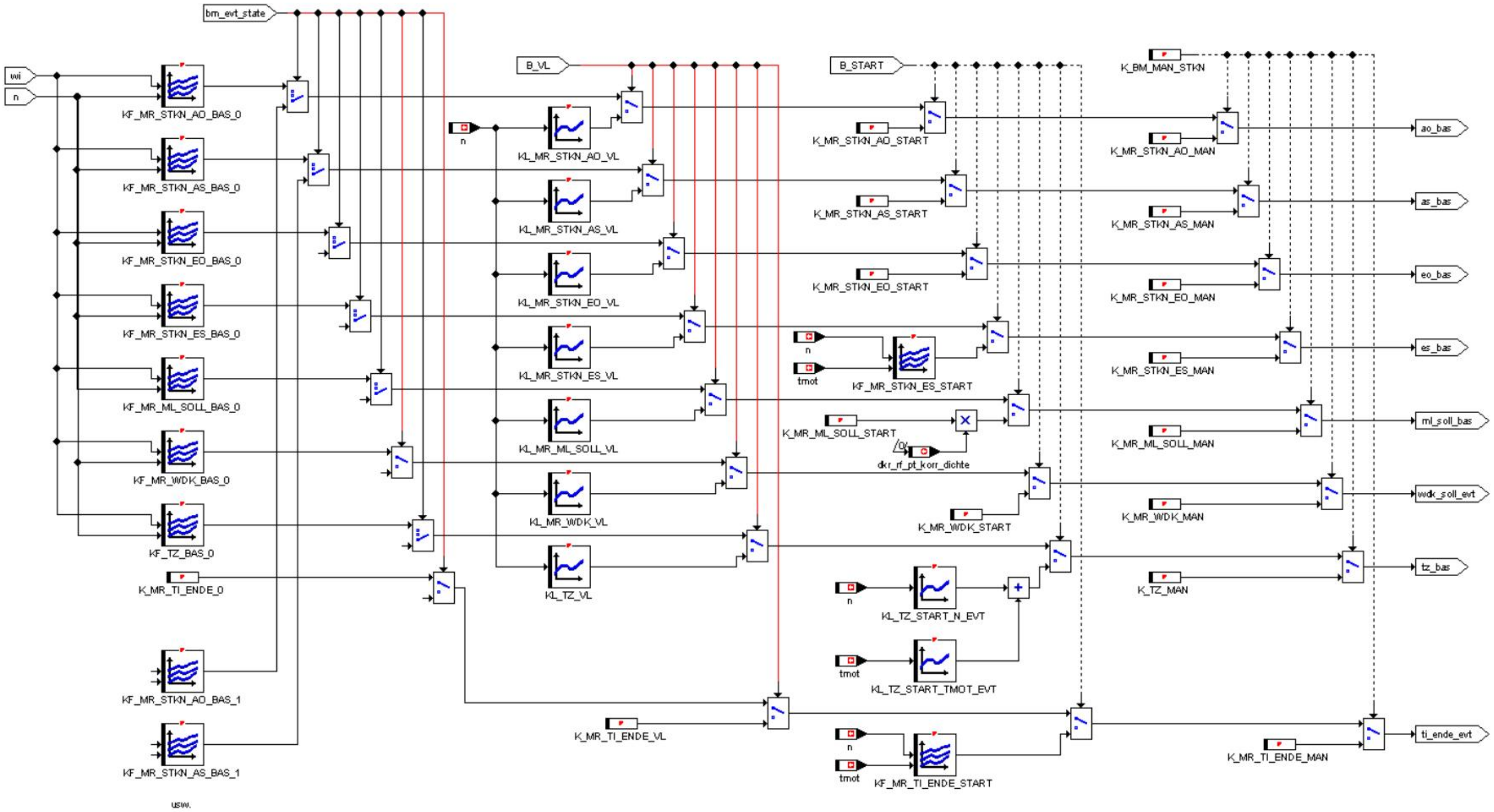
Table of Contents

1 FUNCTIONAL DESCRIPTION

1.1 FUNCTIONAL CIRCUIT DIAGRAM (OVERVIEW)

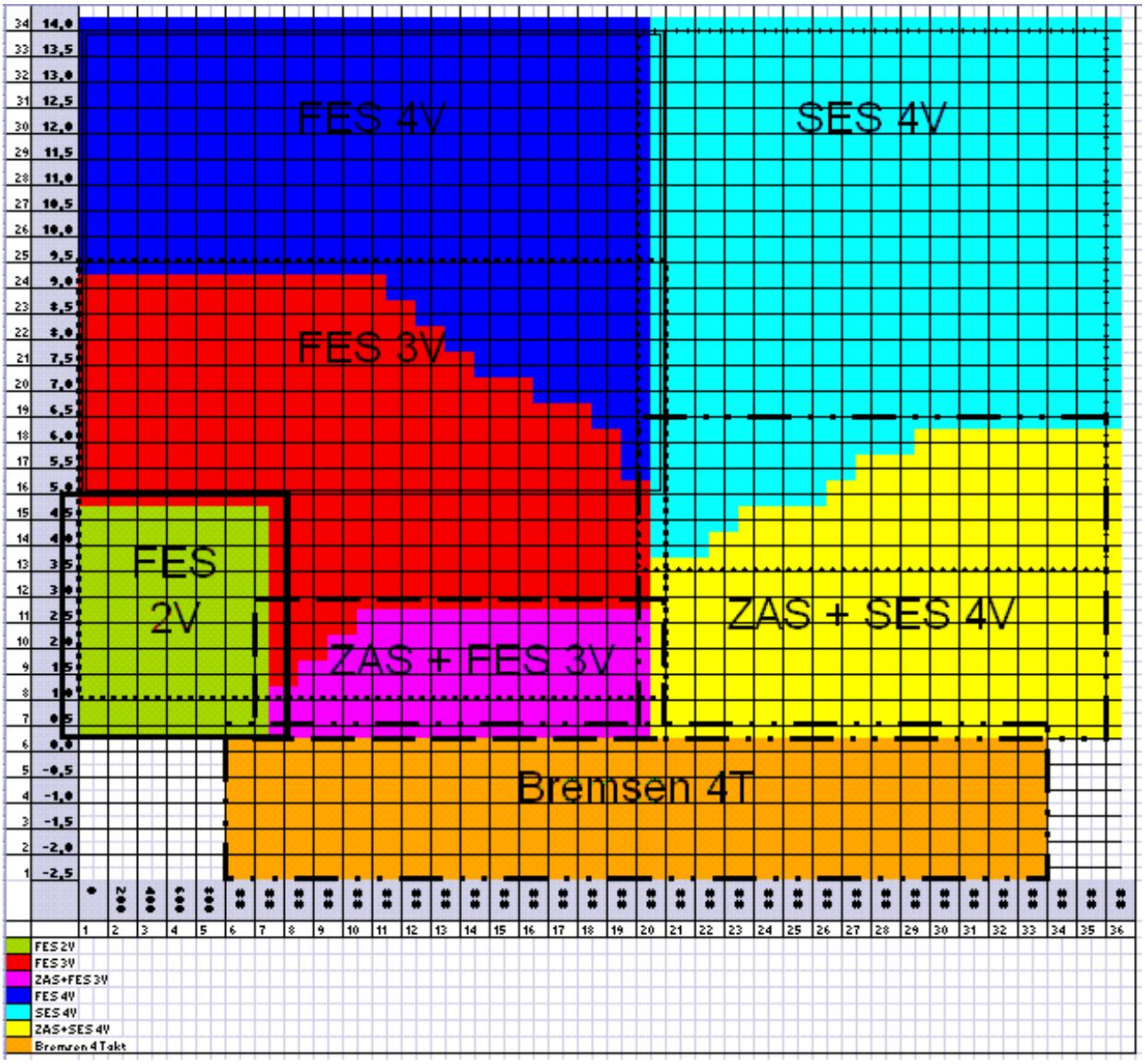


1.2 FUNCTIONAL DIAGRAM BASIC CONTROL EDGES



1.3 DESCRIPTION

The selects according to the applicable operating mode **bm_evt_state** (see operating mode manager).
torque realization the basic maps of this operating mode from:



At full load **(B_VL =** 1) a basic set of characteristics is selected. A separate data record is selected for the start **(B_START =** 1). In addition, a set of control parameters that can be entered manually can be selected via the **B_MAN_STKN** parameter.

The basic control parameter set consists of:

- **eo_bas** (inlet opening control edge in °KW after ignition TDC)
- **as_bas** (exhaust closes control edge in °KW after ignition TDC)
- **es_bas** (inlet closing control edge in °KW after ignition TDC)
- **ao_bas** (exhaust opens control edge in °KW after ignition TDC)
- **wdk_soll_evt** (basic throttle position in %)
- **tz_bas** (basic ignition angle in °CA before ignition TDC)
- **ti_ende_evt** (end of injection in °KW before intake closes)
- **ml_soll_bas** (basic air mass in mg/l*ASP)

The DISA is kept in the power position in all operating modes except full load. At full load, a speed query NMIN_DISA < n < NMAX_DISA decides whether to switch to the torque setting (see Disa.doc).

The control parameters (basic parameters + corrections) are cycle-consistent with the exception of the DISA position and the throttle valve position, ie they belong together for one working cycle of a cylinder (see operating mode manager).

DISA and the throttle valve are synchronized as well as possible with the other cycle-synchronous setting parameters by means of speed-dependent control time offsets.
The basic parameters are stationary at 960mbar and 20°C.
The maps are plotted against **wi** and **n .**

1.4 DO NOT APPLY BIT

In order for the valve control to apply the control edges correctly in every operating mode, a so-called "do not apply bit" (bm_msk_stkn) is set by the MSS54 and transmitted via CAN. This bit encodes which control edges are used and which are not used.
The bit is encoded as follows:

as2	ao2	as1	ao1	es2	eo2	es1	eo1

In the case of cylinder deactivation, for example, the calculated control edges for cylinders 2 and 3 must not be executed; this bit then contains the value 00000000 (00h) for these cylinders.

State 0 1	cylinder 1	Cylinder 2	Cylinder 3	cylinder 4
	FFh	00h 00h	00h 00h	FFh
	3Fh / CFh (180ÿ)			3Fh / CFh (180ÿ)
	3Ch / C3h (720ÿ) 3Ch /	C3h (720ÿ) 3Ch / C3h (7	20ÿ) 3Ch / C3h (720ÿ)	
	3Fh / CFh (720ÿ) 3Fh /	CFh (720ÿ) 3Fh / CFh (7	20ÿ) 3Fh / CFh (720ÿ)	
2 3 4, 5,	FFh	FFh	FFh	FFh
13 6	F0h	F0h	F0h	F0h

In addition, the valves can be completely closed via the **K_MR_VENTZU_EIN** parameter in 4V braking mode **(bm_msk_stkn=0).**

Machine Translated by Google

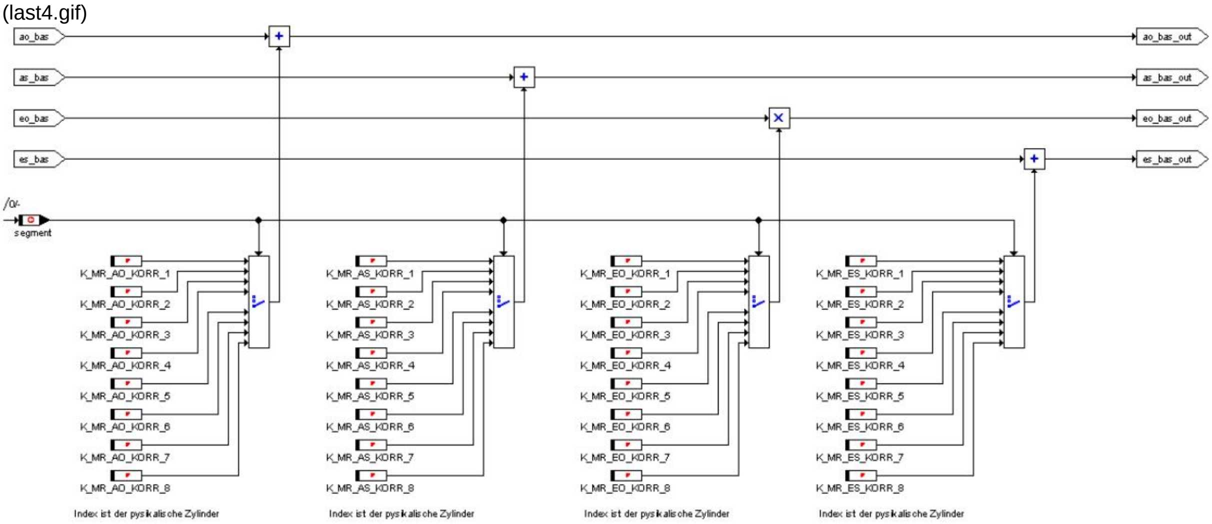
1.5 INDIVIDUAL CYLINDER CONTROL EDGE *CORRECTION*

In order to be able to equalize the cylinder charge and the residual gas content of the cylinder, cylinder-specific control edge corrections are required.

Therefore the 4 control edges (ao_bas, as_bas, eo_bas, es_bas) can be changed with an offset. These offsets, one array each for ao/eo/es, can be set as a manual correction via the application system.

The designation of the arrays is:
K_MR_AO_KORR [1..8]
K_MR_AS_KORR[1..8]
K_MR_EO_KORR [1..8]
K_MR_ES_KORR[1..8]
The index of the arrays refers to the physical cylinder. So: index=1 is for cylinder 1 index 8 for cylinder 8 etc.

(last4.gif)



1.6 INLET CLOSES CORRECTIONS

1.6.1 DENSITY CORRECTION HAS REPLACED BY DKR !

The ambient pressure and temperature that deviate from the standard state are summarized in the factor **rf_pt_korr** and compensated for in an inlet-closes correction.
With constant **wi** and constant AÖ, AS and EÖ control edges, the inlet closes is converted into an actual volume via a volume **characteristic KL_ES_VOLUM .** The density ratio between actual and target density then leads to a new desired air volume. This is converted back into an inlet closing control edge via the inverse characteristic curve **KL_ES_VOLUM_inv .**

This procedure keeps the load point constant in the event of deviating ambient conditions and, in particular, does not change the thermodynamically relevant influencing variables (residual gas, etc.).

The intake closes correction is limited at full load and in the uppermost partial load.

1.6.2 ZW EFFICIENCY CORRECTION (NOT IMPLEMENTED YET!)

In the case of ZW retarded positions, which are caused by knock control and other functions, this is analogous are increased, the air mass over the intake closing edge to compensate for the drop in torque.

This correction is only used with ZW late position, which undesirably reduces the engine torque.

The correction is made using the same characteristic curves. The torque ratio actual torque/maximum torque defined as ignition angle efficiency is determined. The drop in torque is compensated for by an increase in air mass (reciprocal of the torque ratio actual torque/maximum torque).

The resulting control parameter sets keep the moment **wi** constant. The intake-closes correction reduces the residual gas content in the case of retarded ignition angles (tendency to knock is reduced) by keeping the remaining control edges constant.

The intake closing correction due to the retard angle leads to a higher air mass. This is opened in the air mass path via **md_eva_ve .**

1.7 EXHAUST OPENS DELAY

The combusted fuel-air mixture in the cylinder must also be pushed out again with the AÖ control edge, which matches the control edges with which the fresh air was sucked in. The control edge outlet opens therefore thermodynamically belongs to the previous working cycle.

Since the control edges are always calculated in the same segment, AÖ must be delayed by exactly one working cycle (720 degrees CA) in order to then be transmitted to the valve control unit via CAN.

1.8 MINIHUB

The mini lift operating mode is used in the lower load range at low speeds and enables the engine to run quietly.
The amplitude of the control valves is specified by the MSS54, transferred to the dSpace systems via CAN and adjusted there. The minilift is currently only intended for the intake valves, the exhaust valves are operated with full lift in alternating mode (3V) **(mr_minilift_ex =** 0).
The amplitude can be set using the application **constant K_MR_MINILIFT_INT .**
The variable **mr_minilift_int** shows the value of the set valve lift height that is transferred to the CAN. Due to technical program reasons of the dSpace systems, **mr_minilift_int** must be sent to the CAN with a delay of one segment (180 degrees CA).

1.9 AIR MASS ADAPTATION (NOT IMPLEMENTED YET!)

The aim of the air mass adaptation is to compensate for air mass errors in the pre-controlled air mass calculation. A comparison is made between the measured air mass **ml_act_aw and** the pre-controlled air mass **ml_soll_bas .** The difference is fed to an adaptation map via a PT1 filter.

The actual air mass is determined via HFM (ml) and via the lambda probe adaptation **(f_ti_a*ml_soll_bas).** The determination of the actual air mass can be weighted between HFM and lambda probe adaptation via the characteristic curve **KF_FAK_ML_HFM_LAM .**

Adaptionsbedingungen:

- Lambda control is running
- wi below threshold
- B_TL
- Engine at operating temperature

ml_korr_diff_bas < threshold; otherwise error detection

ml_korr_diff_bas = 0 !!!
The air mass adaptation is not implemented at the moment!!! Must be specified in more detail. A separate adaptation map would have to be stored for each operating mode.

1.10 CONVERSION OF ML_SOLL_KORR_EFF INTO INJECTION TIME

The load variable **tl** and from it the injection time **ti** is cycle-consistent from **ml_soll_korr_eff** calculated.

The injection time **ti** is calculated cycle-consistently for each cylinder and each work cycle.

1.11 CONVERSION OF ML_SOLL_KORR_EFF INTO AIR MASS FLOW

The target air mass flow is not required for the basic application. For exhaust gas temperature models or adaptation with the HFM, the target air mass flow can be calculated using the moving average over one working cycle (4 segments with 4 cylinders):

$$[\,m\ l\ kgh\ /\,] \ ÿ \sum_{i\ Segmentnr\ Zylzahl\ (\ \ 1)}^{Segmentnr} (ml\_soll\_korr\_eff\ [\ /\ *_i\ mgl\ ASP\ n\ U\ \ ]\ *\ [/\ min]\ *\ \frac{KRF\ LIFT\ VOLUME\ m^{3}\ ]}{[\ cfg\ number\ of\ cylinders} *0.5*6\ 0/1\ 0\ )^{6}$$

The air mass flow results from the moving average of all cylinders. If the cylinder is switched off, the value 0 is used for **ml_soll_korr_effi .** The air mass flow **ml** is output in [kg/h].

1.12 MASS AIR FLOW TO RELATIVE FILLING CONVERSION

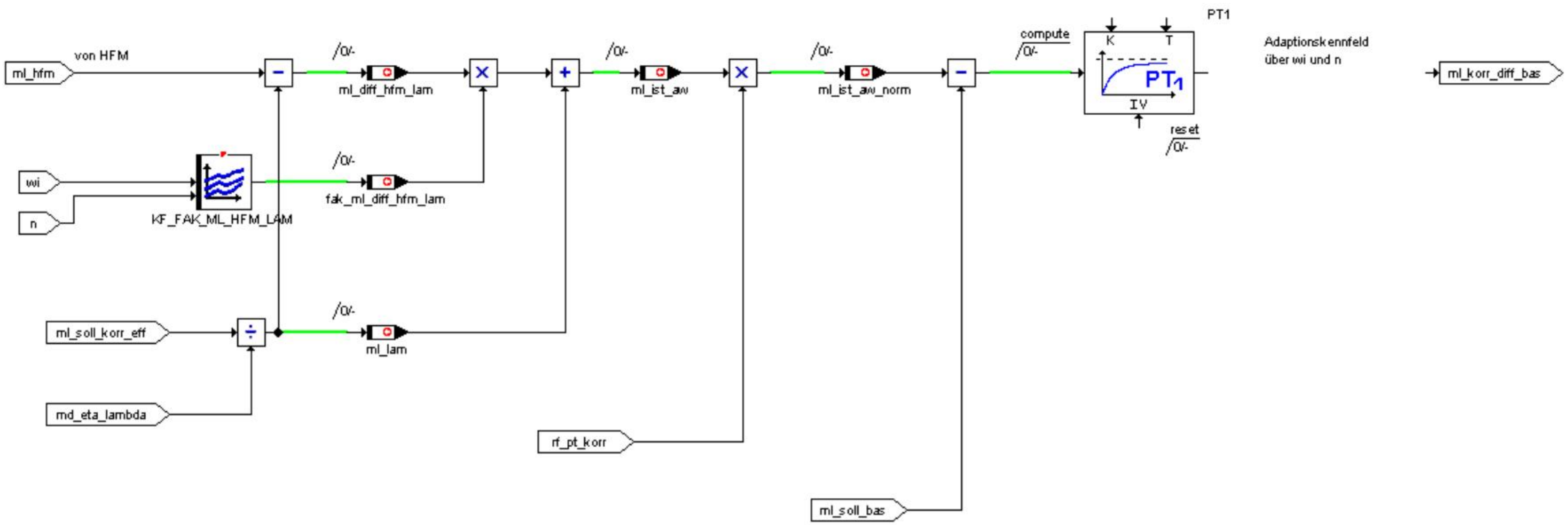
The following formula is used for the conversion to rf:

$$rf\ ÿ\ \frac{ml}{K\_RF\ DISPLACEMENT\ VOLUME\ *K\_RF\_AIR\ DENSITY*0.5*\ \ n}$$

The relative filling **rf** has the unit [%].

Machine Translated by Google

1.13 FUNCTIONAL CIRCUIT DIAGRAM AIR MASS ADAPTATION



2 DATA OF MOMENT REALIZATION

The function is calculated in the angle-synchronous task.

Description of the calculated variables:

ao_aw	Exhaust Opens, current value, delayed by 720 degKW Exhaust Closes,	your
as_aw	current value Inlet Opens, current value Inlet, Closes, base Inlet Closes,	your
eo_aw	current value (density corrected)	your
es_bas		your
es_aw		your
ml_soll_bas	Target air mass, basis [mg/l*ASP]	your
ml_soll_korr	Target air mass, corrected with adaptation Target	your
ml_soll_korr_eff	air mass, corrected with adaptation and ZW air mass from	your
ml_hfm ml	HFM [kg/h]	your
ml_korr_diff_bas	Air mass [kg/h] calculated on basic air mass maps etc	
ml_diff_hfm_lam	Adapted delta target air mass ----------- = 0!!!	your
wdk_soll_evt tz_bas		
ti_ende_evt	Target throttle angle in %	your
bm_msk_stkn	Basic Zundwinkel	sw
mr_minilift_int	Advance angle possibly related to ignition TDC do not	your
mr_minilift_ex	apply bit	where
	Amplitude mini lift inlet	where
	Amplitude mini lift exhaust = 0	where

Description of the application data:

K_TI_ENDE_x	Forward angle at bm_evt_state=x	your
K_TI_ENDE_VL	Forward storage angle for full load operation	your
KF_TI_ENDE_START	Advance angle for takeoff	your/your/your
K_TI_ENDE_MAN	Advance angle for manual mode	your
B_MAN_STKN	Switch to manual mode	where
K_MR_VENTZU_EIN manual closing of the valves only when braking		where
K_STKN_AO_MAN	Outlet Opens for manual mode	your
K_STKN_AS_MAN	Outlet Closes for manual mode	your
K_STKN_EO_MAN	Inlet Opens for manual mode	your
K_STKN_ES_MAN	Inlet Closes for manual mode	your
K_ML_SOLL_MAN	Target air mass for manual mode	your
K_WDK_MAN	Throttle angle for manual mode	your
K_TZ_MAN	Ignition angle for manual mode	sw
K_STKN_AO_START	Outlet Opens for take off	your
K_STKN_AS_START	Outlet Closes for start	your
K_STKN_EO_START	Inlet Opens for start	your
KF_STKN_ES_START	Inlet Closes for start	your/your/your
K_MR_MINILIFT_INT	Amplitude mini lift for intake	where
K_ML_SOLL_START	Desired air mass for takeoff	your
K_WDK_START	Throttle angle for takeoff	your
KL_TZ_START_N_EVT	Ignition angle at start f(n)	your/sw
KL_TZ_START_TMOT_EVT Zündwinkel bei Start f(tmot)		ub/sw
KL_STKN_AO_VL	Outlet opens for full-load operation Outlet	your/your
KL_STKN_AS_VL	closes for full-load operation Inlet opens for	your/your
KL_STKN_EO_VL	full-load operation Inlet closes for full-load	your/your
KL_STKN_ES_VL	operation Target air mass for full-load	your/your
KL_ML_SOLL_VL	operation Throttle valve angle for full-load	your/your
KL_WDK_VL	operation Ignition angle for full-load operation	your/your
KL_TZ_VL	Conversion of inlet closes -> volume inverse	your/sw
KL_ES_VOLUME	characteristic of KL_ES_VOLUM cannot be applied!	your/your
KL_ES_VOLUM_inv		your/your
KF_STKN_AO_BAS_x	Exit Opens at bm_evt_state = x Exit Closes	your/your/your
KF_STKN_AS_BAS_x	at bm_evt_state = x Exit Opens at bm_evt_state	your/your/your
KF_STKN_EO_BAS_x	= x Exit Closes at bm_evt_state = x Should air	your/your/your
KF_STKN_ES_BAS_x	mass at bm_evt_state = x Throttle angle at	your/your/your
KF_ML_SOLL_BAS_x	bm_tw_st_evices_tw_tw	your/your/your
KF_WDK_BAS_x		your/your/your
KL_WDK_BAS_6		
KF_TZ_BAS_x	Basis Zündwinkel bei bm_evt_state=x uw/uw/sw	

PROJECT: MSS54

CHAPTER: 1.05

MODULE: DENSITY CORRECTION IN EVT MOMENTS REALISIERUNG

FUNCTION: DENSITY CORRECTION

AUTHORISATION

AUTHOR (EA-E-2) SCHLUTER DATE 05/03/2004

EDITOR (MSS54) LUBRICANT DATE 12/21/2004

APPROVED (MSS54) ____________________ DATE __________

APPROVED (EA-E-2) ____________________ DATE __________

	Department	date	Name	filename
Author	ZS-M-57	03.05.04	W. Schlueter	1.04

Changes: S380

Version	date	comment
S370	April 30, 2004	1st version as a separate module; Replacement of existing scopes in the EVT moment realization module
S370	05/11/2004	Delivery status Open points: - Lists for operating modes
S370	04.07.2004	Submission stand for the mini-team
S380	12/21/2004	ks: documentation of the implementation

Table of Contents

1 FUNCTIONAL DESCRIPTION

The DKR density correction control module contains control functions that compensate for the influence of a changed intake air condition. However, intake pressure and temperature are included in the calculation differently. The compensation prevents a changed intake air condition from leading to a change in fresh air charge, residual gas content, charge movement and consequently to a change in the indicated work at the corresponding operating point.

Operating altitudes between -300 and +3000 m above sea level. correspond to pressure changes between +10% and -30% compared to a reference pressure of 960 mbar. For a reference temperature of 293 K, there are temperature changes of approx. +-10% in the relevant operating range.
The inflow behavior, which is particularly decisive for the fresh air filling, is primarily determined by the air condition in the intake manifold. The mean intake manifold pressure is therefore used as the input variable for calculating compensation measures.
The method implemented here is purely a correction of the inlet closes control edge with the aim of adapting the fresh air charge to the applied value under reference conditions. The warming of the gas before it is pushed out again at Late Admission Closes is not explicitly taken into account.
Assuming that the outflow process is primarily influenced by the combustion, i.e. the indicated work, and less by the ambient condition, the residual gas mass in the cylinder is not corrected. The influence of charge motion is neglected.

The two calculation methods for the intake closes correction ÿ based on the cylinder volume at intake closes or the opening time of the intake valve ÿ and the subsequent limitation of the corrected intake closes control edge are described in more detail below.

1.1 PHYSICAL BACKGROUND

The intake closes correction uses two parallel calculation methods: With the focus on part-load operation with full lift of the valves, i.e. for operating points in which the fresh air filling is limited by the cylinder volume, the cylinder volume is evaluated when the intake closes. Assuming that the gas density in the cylinder is proportional to the ambient condition at this point in time, the inlet closing control edge is shifted in such a way that the product of density and cylinder volume at inlet closing is equal to the applied reference state. With the cylinder *volume VES* as a geometric function of the intake closes crank angle and the relative air density in the intake manifold *rf_pt_korr_dicht ,* the following applies:

$$VES,korr = rf\_pt\_korr\_dichte\ VES,ref = \frac{\left(p/T\right)_{ref}}{p/T}\ V_{EN\ reference}$$

With early intake closing, a reduced density leads to a larger cylinder volume, ie later intake closing.

This correction corresponds to the correction function up to control unit version R 360.
If the intake closes late and the density is reduced, the necessary larger cylinder volume is achieved by closing the intake earlier. Cylinder volume as a function of crank angle is symmetrical to bottom dead center at 540°. To expand the application options, however, the Inlet Closes control edges are transformed to Late Inlet Doesn't Close [with ES := 1080 - ES] in the Early Inlet Closes area. Instead, the cylinder volume function is stored separately for this area.

The opening time of the intake valve is evaluated with the focus on mini lift, i.e. for operating points in which the fresh air filling is determined by the inflow behavior of the intake valves. For early inlet close modes, this is the distance between inlet open and inlet close. For operating modes with late admission Opens instead of admission Opens the start of the re-expulsion phase is relevant; this time corresponds approximately to the bottom dead center. With the relative intake mass flow *rf_pt_korr_drossel* follows for the opening time of the intake valve:

$$ESkorr - EO = (ESref - EO)\ rf\_pt\_korr\_drossel$$

The throttle characteristic or a laminar-turbulent approach for the intake mass flow can be stored in two characteristic curves for the dependency on pressure and temperature:

$$rf\_pt\_korr\_drossel = \frac{\dot{m}}{\dot{m}_{ref}} \text{ ÿ } f\ ÿpÿ\ f\ ÿT\ ÿ$$

A weighted average of both correction models is used for operating points with high speeds or loads. The specific load per cylinder and intake valve is used as input to the map to weight the opening-time-based correction.

Before the calculation of the volume-related inlet closing correction, the inlet closing control edge can be shifted compared to the calculation of the cylinder volume. This allows dynamic effects (pressure waves, resonances) to be taken into account. As an alternative to the proportionate weighting of the opening-time-based correction, the inflow pressure losses at high loads and speeds can also be taken into account with this intervention.

After the corrected intake closes control edge has been calculated, it is limited to the physically meaningful range: Depending on the early or late intake closes operating mode, the limits here are the dead centers of the piston movement or the full-load control times.

1.2 IMPLEMENTATION

In the signal flow, the density correction module converts the intake closes control *edge es_bas* formed in the EVT torque realization module from the basic maps or the application intervention into a corrected intake closes control *edge drk_es_aw* (previous name: *es_aw).*
The relative density *rf_pt_korr* is also provided for external calculations. This is set equal to the relative density for the volume-related inlet closes correction *drk_rf_pt_korr_dicht.* The relative density *rf_pt_korr* is also provided for external calculations. The relative flow *drk_rf_pt_korr_drossel* is also used externally. All other variables calculated in this module are internal.

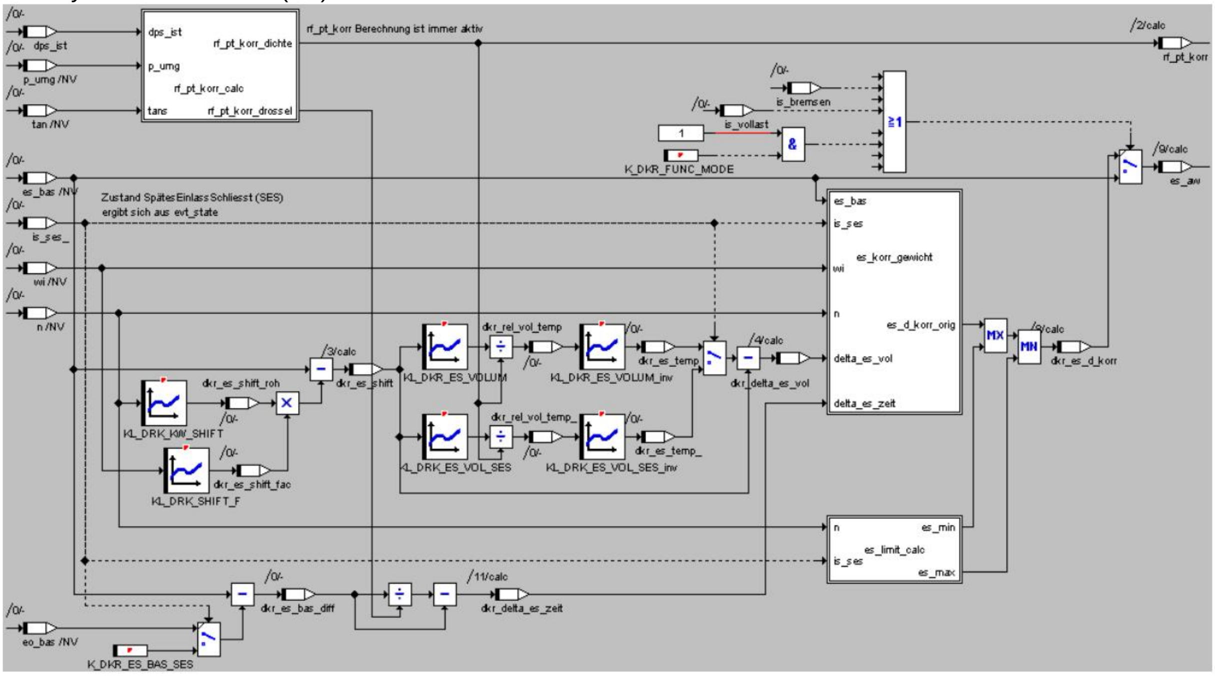
With the exception of the KL_STKN_ES_VL characteristic, all parameters are internal to the module.

Note on realization:

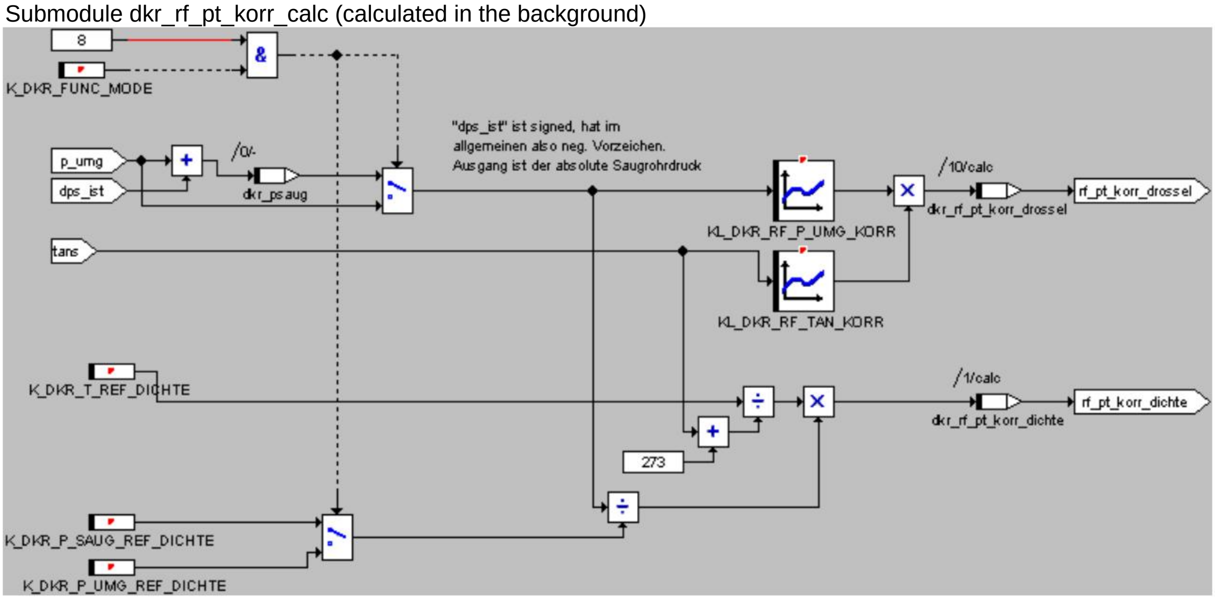
The function is very time-consuming, since large parts of it (with a number of interpolations) are calculated in the segment grid. If the MSS54 is operated on an 8-cylinder engine, it is advisable to switch off the density correction (K_DKR_FUNC_MODE = DKRoff), as otherwise the performance is not sufficient for higher speeds. It may be necessary to reconsider the design of the function in the future in order to get by with less computing time or computing frequency.

1.3 FUNCTIONAL DIAGRAM

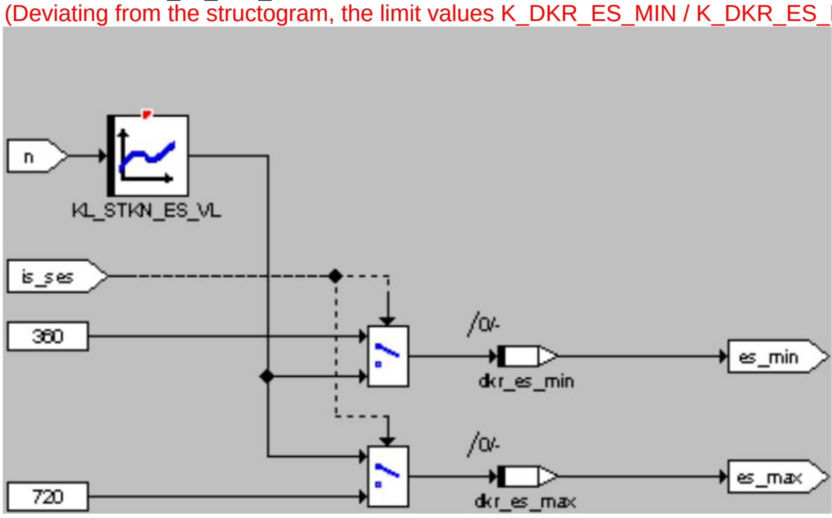
Density correction module (dkr)



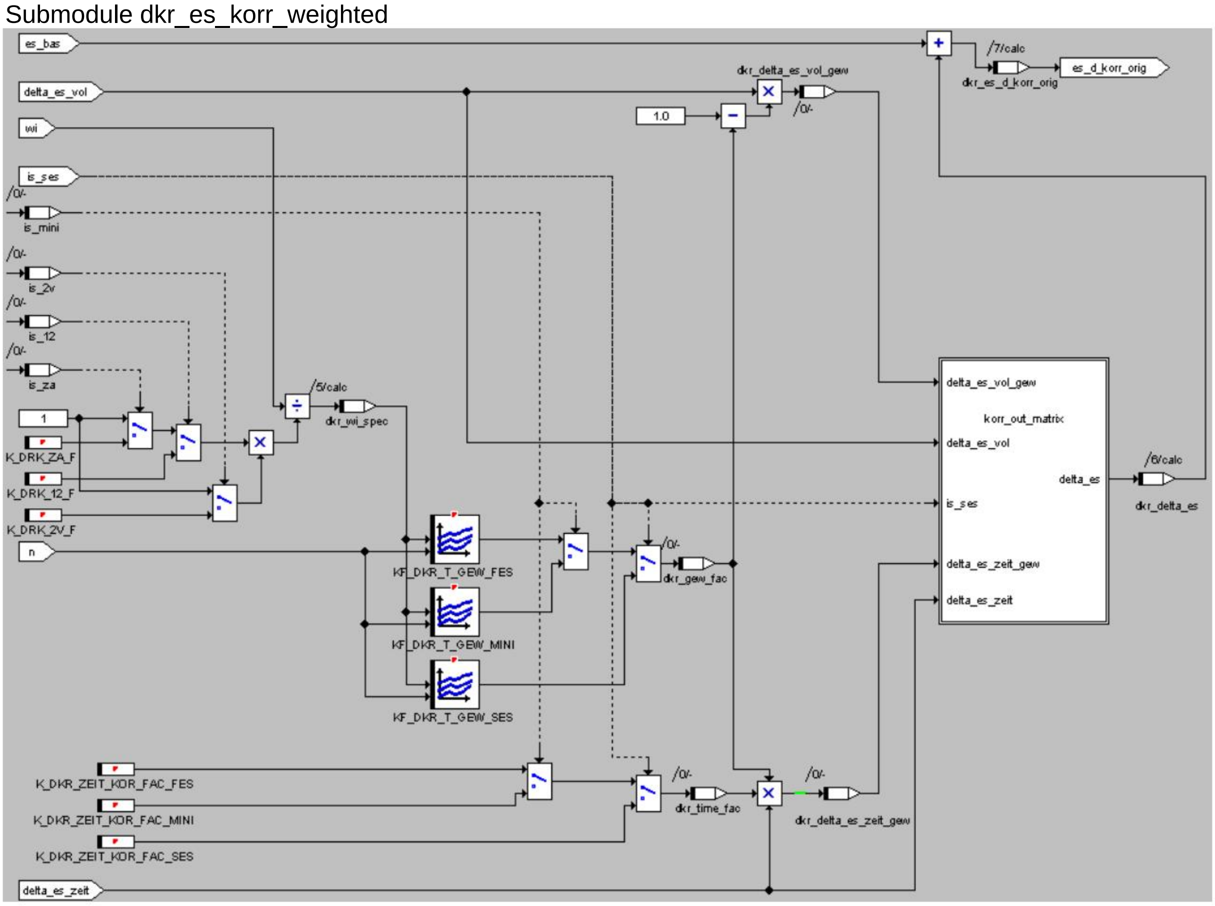
Submodule dkr_rf_pt_korr_calc (calculated in the background)



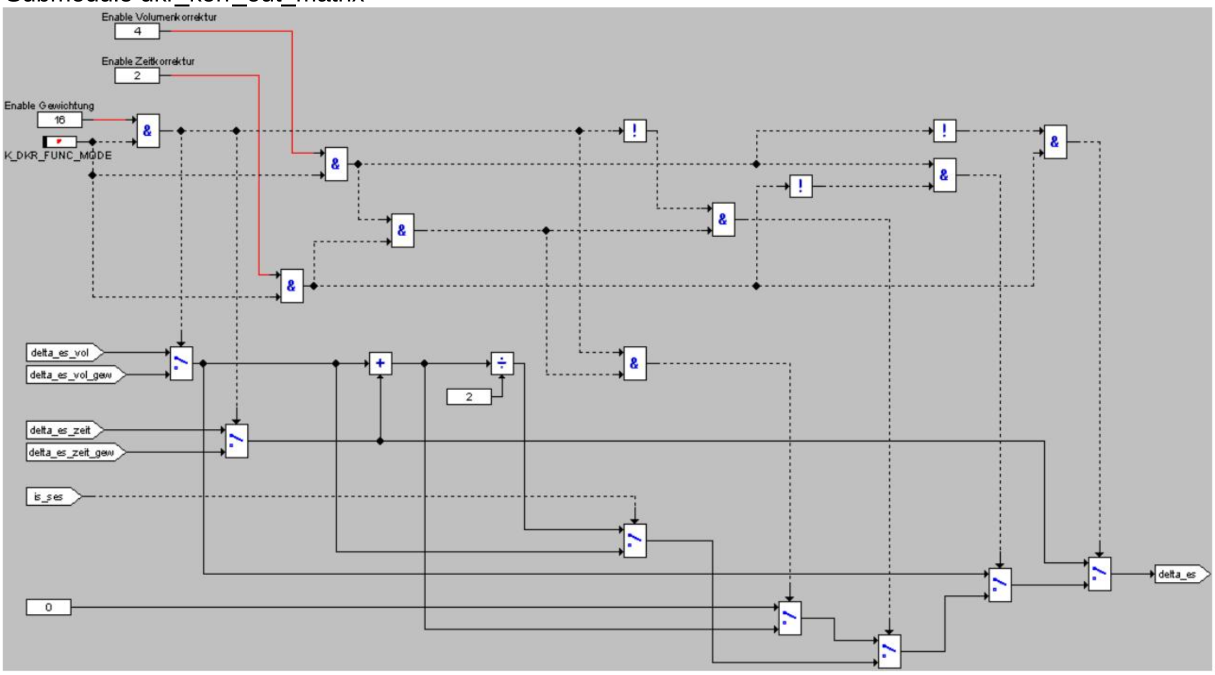
Sub-module dkr_es_limit_calc
(Deviating from the structogram, the limit values K_DKR_ES_MIN / K_DKR_ES_MAX are used in "dkr_es_limit_calc".)



Submodule dkr_es_korr_weighted



Submodule dkr_korr_out_matrix



1.4 APPLICATION NOTES

The reference ambient condition is 960 mbar at 20°C. With an intake manifold vacuum of 50 mbar at most operating points, the reference intake manifold state has an air pressure of 910 mbar.

2 MODULE DATA

The function is calculated segment-synchronously in the slave.
rf_pt_korr Calculations take place in the background

	Shop dkr	background	1ms	10ms	20ms	100ms	1s
Task		dkr_rf_pt_korr_calc					

2.1 VARIABLES

The module does not contain any static variables, all sizes are global.

Variable [Output]	Initialization unit area		(physical.)	How much	Impl.	page
dkr_es_aw		°	0 - 720	0,1	word	
	Crank Angle Inlet Closes Out Density Correction					
	global output variable					
rf_pt_korr		-	0 - 2,5	x/128	byte	
	1 rf_pt_korr to external functions set as rf_pt_korr _density					
	Calculated from: p_umg, dps_actual, tan					
dkr_rf_pt_korr_drossel	1 byte	-	0 - 2,5	x/128		
	rf_pt_korr (relative mass flow) for density correction via intake valve opening time					
	Calculated from: p_umg, dps_actual, tan					

Variable [Local]	initialization	unit	Area (physical.)	How much	Impl.	page
dkr_rf_pt_korr_dichte	1	-	0 - 2,5	x/128	byte	
	rf_pt_korr (relative density) for density correction over cylinder volume					
dkr_delta_es_vol	0 0,1	°	-180 - 180		word	
	Inlet Closes crank angle correction from cylinder volume					
dkr_delta_es_time	0 -180 - 180 0,1	°			word	
	Intake Closes crank angle correction Intake valve opening time					
dkr_delta_es	0 0,1	°	-180 - 180		word	
	Inlet Closes Crank Angle Correction					
dkr_es_d_korr_orig		°	0 - 720	0,1	word	
	Inlet crank angle Closes after density correction without min/max limitation					
dkr_es_d_korr		°	0 - 720 word	0,1		
	Crank angle intake closes calculated in density correction (return value from "dkr()")					
dkr_es_min		°	0 - 720	0,1	word	
	minimum value limitation					
dkr_es_max		°	0 - 720	0,1	word	
	maximum value limitation					
dkr_es_shift		°	0 - 720	0,1	word	
	Work value in "dkr()"					
dkr_wi_spec		kJ/l	who "wi"		word	
	Work value in "dkr_es_korr_weighted()"					
dkr_gew_fac			0 - 1	0,05	byte	
	Work value in "dkr_es_korr_weighted()"					
dkr_time_fac			0 – 12,7	0,05	byte	
	Work value in "dkr_es_korr_weighted()"					
dkr_delta_es_zeit_gew 0		°	-180 - 180	0,1	word	
	Work value in "dkr_es_korr_weighted()"					
dkr_delta_es_vol_gew	0	°	-180 - 180	0,1	word	
	Work value in "dkr_es_korr_weighted()"					

Variable [Input]	Those	unit	Area (physical.)	How much.	Impl.	page
eo_bas	EVT-Momentary Real.	Deg				
	Control Edge Inlet Opens					
es_bas	EVT-Momentary Real.	Deg				
	Base control edge inlet closes					
evt_state	EVT-Momentary Real.	-				
	operating mode					
so		°C				
	intake air temperature					
p_umg		mbar				
	ambient pressure					
wi		kJ/l				
	indexed work					
n		Rpm				
	number of revolutions					
dps_ist		mbar				
	Manifold vacuum (averaged)					

2.2 PARAMETER

Application size	Default value	unit	area (physical.)	quant.	implement	page
K_DKR_FUNC_MODE	DKR OFF	-	0x00: DKR=0 (ineffective) 0x07: DKR[t/V/ups/Gew]=1 0x13: V/Wt=0 0x15: t/wt=0 0x17: wt=0 0x03: V=0 0x05: t=0 0x0F: ups=0 0x1B: V/ups/wt=0 0x1D: t/ups/Gew=0 0x1F: ups/wt=0 0x0B: V / ups = 0 0x0D: t/ups=0 0x80: DKR OFF (disabled)	-	byte	
	Switch: Inlet Closes Disable/toggle density correction intervention					
K_DKR_ES_BAS_SES	540 byte	°	500 - 755	1		
	Crank angle: start of outflow at SES					
K_DKR_ES_MIN	400	°	500 - 755	1	byte	
	Limit value in "dkr_es_limit_calc" (deviating from the structure chart)					
K_DKR_ES_MAX	660	°	500 - 755	1	byte	
	Limit value in "dkr_es_limit_calc" (deviating from the structure chart)					
K_DKR_P_REF_DENSITY	910	mbar	850 - 1105	1	byte	
	Reference manifold pressure for air condition					
K_DKR_ZEIT_KOR_FAC_FES 1 0.05		-	0 - 12,7		byte	
	Additional weighting factor: opening time-related admission closes correction for FES					
K_DKR_ZEIT_KOR_FAC_Mini 1 0.05		-	0 - 12,7		byte	
	Additional weighting factor: Opening time related inlet closes correction for minihub					
K_DKR_ZEIT_KOR_FAC_SES 1 0.05		-	0 - 12,7		byte	
	Additional weighting factor: opening time-related admission closes correction for SES					

K_DKR_T_REF_DENSITY	293	K 270 - 525		1	byte	
	Reference temperature for air condition					
K_DRK_ZA_F	2	-	0 - 5	0,02	byte	
	Multiplication factor for cylinder load at cylinder deactivation					
K_DRK_2V_F	2	-	0 - 5	0,02	byte	
	Multiplication factor for cylinder load at 2V operation					
K_DRK_12_F	3	-	0 - 5	0,02	byte	
	Multiplication factor for cylinder load in 12-stroke operation					

2.3 CHARACTERISTICS

Application size	provide support	unit area		quant.	implement	page
KL_DKR_ES_VOLUME	8 x KW	°	465 - 720	1	16 * byte	
		.	0 – 2 (3)		16* byte	
	cylinder volume = f(crank angle); Characteristic must be invertible					
KL_DKR_ES_VOL_SES	8 x KW	°	465 - 720	1	16 * byte	
		.	0 - 1		16 * byte	
	cylinder volume = f(crank angle) for SES; Characteristic must be invertible					
KL_DKR_RF_P_UMG_KORR 8 xp		mbar 600 - 1110 8 byte		2	*	
		.	0 - 2,5	x/128	8 * byte	
	KL_DKR_ES_VOLUM stored inversely					
KL_DKR_RF_TAN_KORR	8 xt ° C -40 - 85			1	8 * byte	
		.	0 - 2,5	x/128	8 * byte	
	KL_DKR_ES_VOLUM stored inversely					
KL_DKR_KW_SHIFT	32 x KW	RPM 0 - 7500		50	32 * byte	
		you	-30 - 120	x/128	32 * byte	
	Crank Angle Shift Inlet Closes Cylinder Volume Calculation					
KL_DKR_KW_SHIFT_F	8 x wi kJ/l 8 byte		0 - 1,5	0,01	*	
		.	0 - 2,5	0,01	8 * byte	
	Load dependent crank angle shift weighting					
KL_STKN_ES_VL		Rpm				
		you				
	Base control edge inlet closes full load (included from the load module)					

2.4 MAP

Application size	base n	unit area		How much	. Impl.	page
KF_DKR_T_GEW_FES		U / min	0 - 6500 0		8 * byte	
		kJ / l	- 1,5 0 - 1		8 * byte	
		.		0,05	8*8 * byte	
	Weighting factor for opening time based density correction for FES					
KF_DKR_T_GEW_MINI		U / min 0 - 6500 kJ / l 0 -			8 * byte	
		1.5 0 - 1			8 * byte	
		.		0,05	8*8 * byte	
	Weighting factor for opening time based density correction for minihub					
KF_DKR_T_GEW_SES		U / min 0 - 6500 8 kJ / l 0 - 1.5 0 - 1			* byte	
					8 * byte	
		.		0,05	8*8 * byte	
	Weighting factor for open-time based density correction for SES					

3 INITIAL DATA

In the following, initial data is given for all application values. Additional values are given for some parameters in order to realize the functionality of the old density correction (R360): In this case, the opening-time-based correction as well as the shift between intake closes crank angle and cylinder volume calculation are neutralized.

3.1 PARAMETER

K_DKR_B_DRK_OFF	0	
K_DKR_ES_BAS_SES	540 °	
K_DKR_ES_MIN	400 °	
K_DKR_ES_MAX	660 °	
K_DKR_P_REF_DENSITY	910 mbar (= 960 - 50)	
K_DKR_T_REF_DENSITY	293 K	
K_DKR_ZEIT_KOR_FAC_FES	1	for stand R360: 0
K_DKR_ZEIT_KOR_FAC_Mini	1	for stand R360: 0
K_DKR_ZEIT_KOR_FAC_SES	1	for stand R360: 0
K_DKR_ZA_F	2	
K_DKR_2V_F	2	
K_DKR_12_F	3	

3.2 MAP

KF_DRK_T_GEW_FES:	constant 0
KF_DRK_T_GEW_SES:	constant 0
KF_DRK_T_GEW_MINI:	constant 1

3.3 CHARACTERISTICS

KL_DKR_ES_VOLUME

Characteristic must be invertible

From 540 to 543 two interpolation points with slope 1! Data according to current status, linearly extrapolated

KW 360 390 460 540 Output [] 0.088 0.097 0.168 0.372 0.696 0.925 1					500		720
								3

Calibration from the old software version without DKR.

KL_DKR_ES_VOL_SES

characteristic must be invertible; Data analogous to current status, mirrored at 540°, linearly extrapolated

KW	540	550 630 690 570 0,991 0,925 600 0,761 0,469 0,260 0,130 660 0,088						720
output [ ]	1							

Calibration from the old software version without DKR.

KL_DKR_RF_P_UMG_KORR

P_UMG	599	749	800	851	899	959	1040 1100	
Exit [-]	0,62	0,78	0,83	0,88	0,94	1	1,08 1,14	

KL_DKR_RF_TAN_KORR

SO	-40	-20		20	40	60	80	100
Output [Nm] 1.26		1,16	0 1,07	1	0.94	0,88	0,82	0,73

KL_DRK_KW_SHIFT

Calculated as full load inlet closes - 540° with non-negative limitation.

If driving at full load with a different DISA position, modify the values.

N 400		800	1200 1600 2000 2400 2800				
Output[ ] 0		0	0 12		7	13	20

3200 3600	4000 4400	4800 5200	5600 6000	6400				
26 44 51	34		46		60	74	93	120

In order to implement status R360, this or the characteristic KL_DRK_KW_SHIFT_F must be set to a constant 0. Had to be reduced to 16 support points!

KL_DRK_KW_SHIFT_F

wi 0		0,2	0,4	0,6	0,8	0,9	1	1,4
Output[ ] 0		0,2	0,4	0,6	0,8	0,9	1	1

Project: MSS54

Module:Fill Controller

	Department	date	Name	Filename
Processor	ZS-M-57	05.07.2005	.Frank	FR.DOC

Table of Contents:

1 filling regulator

The filling controller ensures the stationary comparison of the actual filling with the target filling. The filling controller is a PI controller, whereby the I component is switched off (current value is frozen) when the throttle valve is open so far that the engine is no longer throttled, or when the current throttle valve position deviates from the setpoint specification for the position controller is greater than an applicable constant. The P component is set to zero if the condition B_WDK_KEINE_DROSSEL is active.

The controller influences the manipulated variable md_rf_soll multiplicatively.

The controller is supported by a prediction of the filling to be expected in the next step.

1.1 Calculation of the control difference

The controller deviation is calculated as follows:

fr_rf_delta = 10 * rf - md_rf_roh

The factor 10 results from different normalization.

1.2 Predictor

The predictor coefficient is calculated as follows:

fr_rf_prae = kls_wint(&KL_FR_PRAE, n) * (md_rf_roh - md_rf_roh-1)

The controller setpoint is influenced by the predictor coefficient.

fr_rf_delta = fr_rf_delta + fr_rf_prae

1.3 PI - Rules

The charge controller is a PI controller, where the P component is set to zero when the engine is no longer throttled. The I component is set to zero if B_ML is not set or with B_HFM_FEHLER .

The I component is frozen when the engine is no longer throttled and the actual charge is less than the setpoint charge. In addition, the I component is frozen if the deviation of the current throttle valve position from the setpoint specification for the position controller is greater than an applicable constant.

1.4 Data of the filling controller

Machine Translated by Google

Description of the variables:

Name	Description	type resolution
fr_rf_delta	Controller deviation	sw 1/10 000
rf md_rf_roh	relative charge is	your 1/1000
fr_rf_gradient	relative charge should, not p/t corr.	your 1/10000
	change of the rel. filling	your 1/10000 / 10 ms
md_rf_soll	p/t corr. rel. Filling should	your 1/10000
lls_eml.rf_rel_korr	be p/t corr. rel. Filling output I	your 1/10000
fr_reg_i fr_ant_i	component of the filling controller	sw 1/32768
fr_reg_p	P component of the filling controller	sw 1/32768
fr_regler		
fr_rf_prae		
fr_rf_roh_pre		

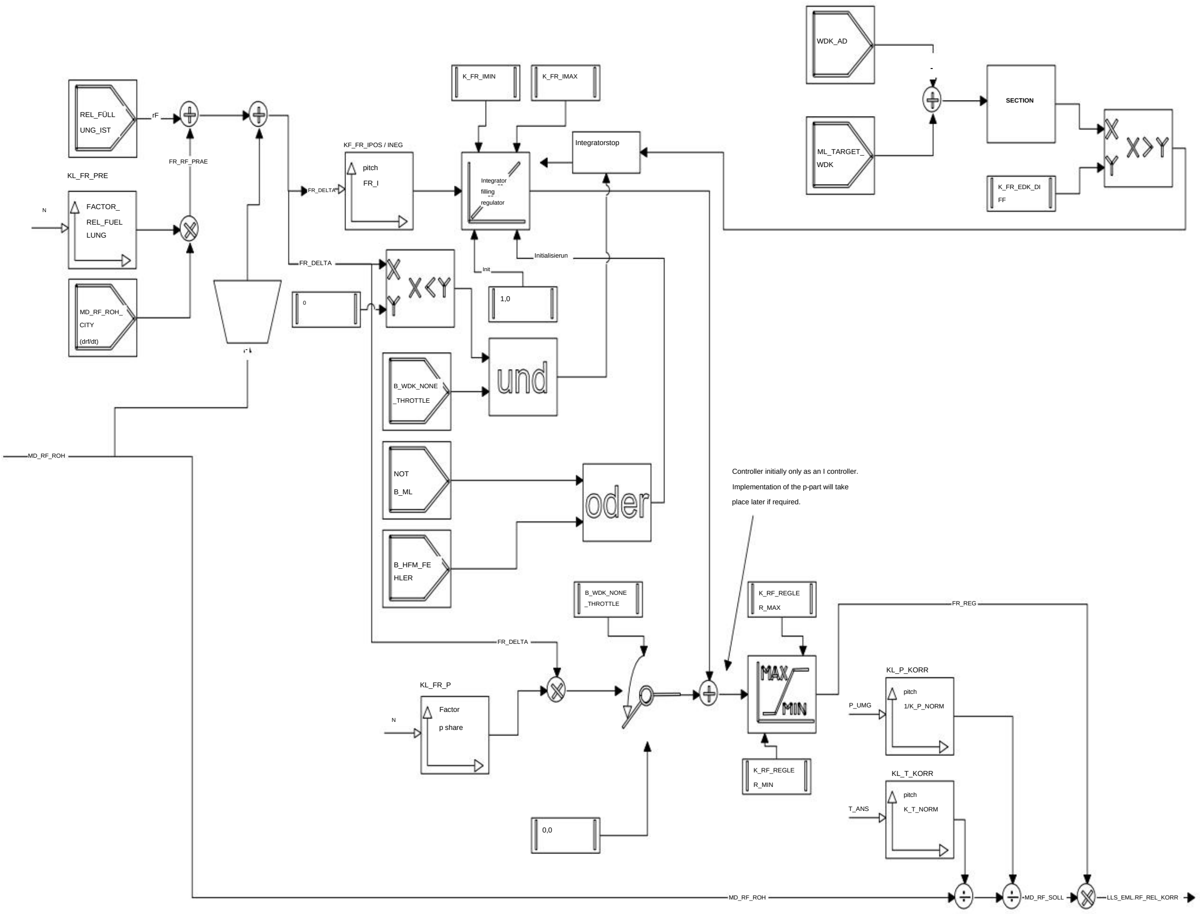
Description of the application data:

Name	Type	Dim. x-axis	y-axis
KL_FR_IPOS	KL	8 x 6 rf Control deviation n_mot	
KL_FR_INEG	KL	8 x 6 rf Control deviation n_mot	
KL_FR_IPOS	KL	8 x 6 rf Control deviation n_mot	
KL_FR_P	KL		
KL_FR_PRE	KL		
K_FR_ADAPT_TOL			
K_FR_DI_ENTDR			
K_FR_DMLADAPT_MAX			
K_FR_EDK_DIFF			
K_FR_IMAX			
K_FR_IMIN			
K_FR_MLADAPT_MAX			
K_FR_MLADAPT_MIN			
K_FR_MLADAPT_OFFSET			
K_FR_TAU_ADAPT			
K_FR_TMOT_ADAPT			
K_FR_T_ADAPT			

1.5 Circuit diagram of the charge controller

Fill control/density correction

Clipper ZS-E-51



Project: MSS54

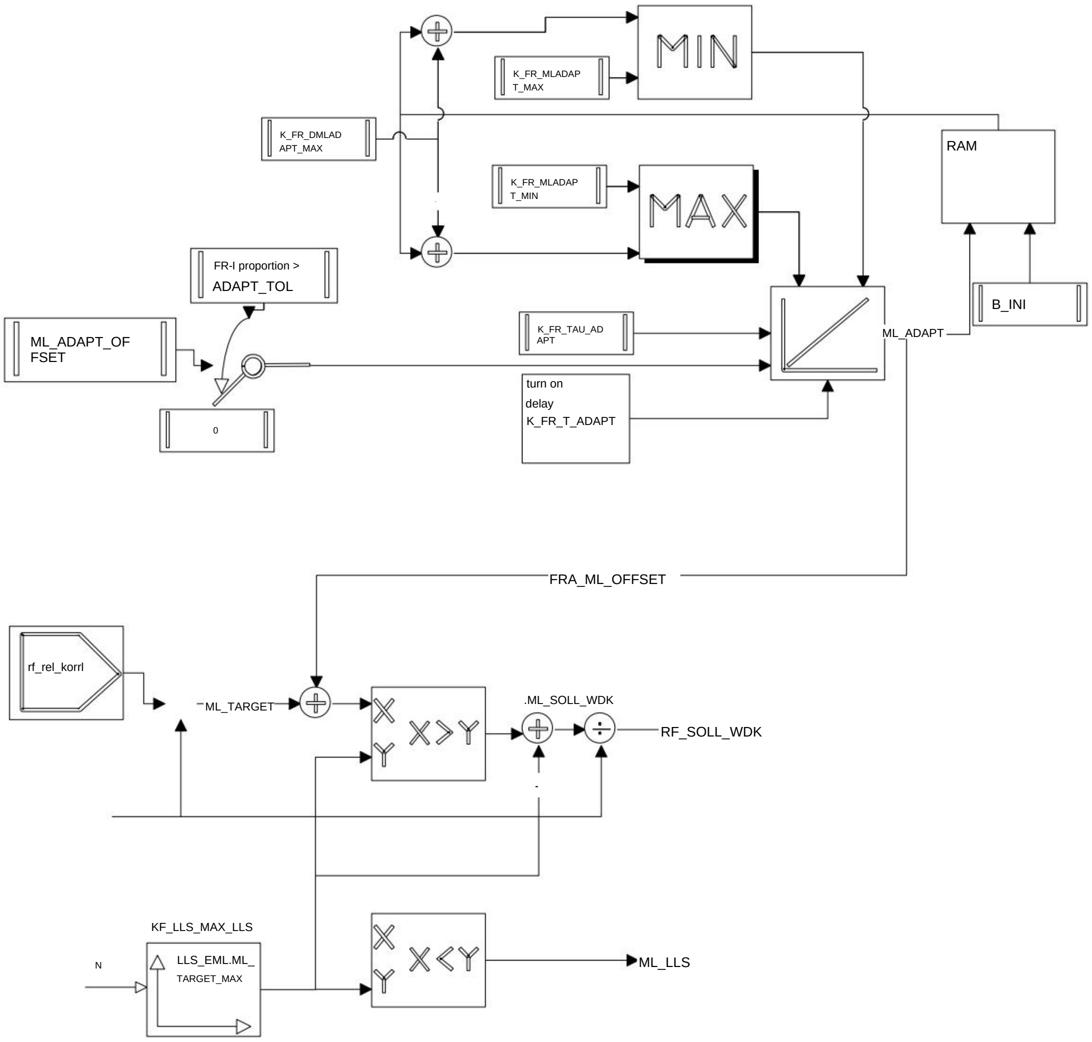
Module: Adaptation of filling controller

Machine Translated by Google

1 Adaptation filling controller

The task of the filling controller adaptation is the **stationary** caused by assembly and production fluctuations of the throttle flaps (different leakage air in different vehicles).
Compensate for differences between the calculated target guidance and the actual filling measured using HFM. This deviation is to be determined by the FR adaptation and rectified by ***correcting the calculated ml_soll .***

Image : Fill regulator adaptation



1.1 Adaptionsbedingungen

To activate the adaptation, the following conditions must be met:

B_FRA = B_LLR ; State idle control active (see state machine of the LLR)
and tmot > K_FR_TMOT_ADAPT ; Engine temperature greater sill
and !B_TMOT_FEHLER and ! ; error-free tmot capture
B_HFM_FEHLER and ! ; error-free ml acquisition
B_KATH_AKTIV and no error ; Cat heating not active
in the EGAS system

During the development and test phase, the complete FR adaptation can be switched off using the K_FRA_CONTROL control byte. All adaptation values are then equal to zero.

1.2 States of FR adaptation

The control of the FR adaptation can be described as a state machine.

Adaptation inactive

Condition: B_FRA not fulfilled

Mark: fra_flags = 0 (inactive)

Adaptionswerte: fra_mladapt (t) = fra_mladapt (t - 20 ms)

Blocking time monitoring for FRA active

Condition: B_FRA met
und fra_timer! = 0
(Locking time not yet expired

Mark: fra_flags = 1 (Block time)

Adaptionswerte: fra_mladapt (t) = fra_mladapt (t - 20 ms)

Adaptation running

Condition: B_FRA
and fra_timer == 0 (blocking time expired)
und | fra_mladapt - fra_mlstart | ÿ K_FRA_DMLADAPT_MAX
(adaptation path not limited)

Mark: fra_flags = 3 (adapted)
adaptation values: fra_mladapt (t) = fra_mladapt (t - 20 ms) +
K_FR_MLADAPT_OFFSET) * K_FR_TAU_ADAPT
(without considering a limitation)

Adaptation value limited

Condition: B_FRA
und fra_timer == 0
und | fra_mladapt - fra_mlstart | > K_FR_DMLADAPT_MAX
(Limited adaptation path)

Mark: fra_flags = 7 (limited)

Adaptionswerte: fra_mladapt (t) = fra_mlstart ÿ K_FR_DMLADAPT_MAX

Annotation: If the difference between the calculated adaptation value and the start value at the beginning of the adaptation phase is again smaller than the maximum adaptation path, you change back to the "adaptation running" status.

1.3 Data of the FR adaptation

Description of the variables:

Name	description	Type	resolution
fra_timer	remaining adaptation blocking time	your	0,02 sec.
fra.mladapt	Value of the adaptation integrator	SW	Def_rf * 320
fra.mlstart	Value of the adaptation integrator at the beginning of a new one Adaptionsphase	SW	Def_rf * 320
fra.flags	Flags for adaptation Value 0: Adaptation inactive Value 1: Blocking time running Value 3: adapted Value 7: Adaptation path limited	etc.	..
fra_ml_offset	Output variable of the FR adaptation		Def_rf
fra_sperren			

Description of the application data:

Name	Type	importance
K_FR_MLADAPT_OFFSET	FW	Adaptionsoffset
K_FR_ADAPT_TOL	FW	Tolerance band of the filling controller, below no adaptation ie OFFSET = 0
K_FR_TAU_ADAPT	FW	Time constant for adaptation
K_FR_DMLADAPT_MAX	FW	max. Adaptionsweg pro Adaptionsphase
K_FR_T_ADAPT	FW	Adaptionssperrzeit
K_FR_MLADAPT_MIN	FW	Lower adaptation value limitation Upper
K_FR_MLADAPT_MAX	FW	adaptation value limitation Control variable
K_FRA_CONTROL	FW	(adap on/off)
K_FRA_RF_ABREG	FW	rf clamp for curtailment d. Offsets Factor for
K_FRA_RF_FAKTOR	FW	derating d. offsets from b. rf

1.4 Non-volatile storage

In the follow-up phase of the control unit, the current value

fra_mladapt

of the FR adaptation is stored non-volatile in the E²PROM of the control unit

PROJECT: MSS54

MODULE: DIFFERENTIAL INTAKE SYSTEM

AUTHORISATION

AUTOR (EE-221)________________________ DATE __________

APPROVED (ZS-M-57)___________________ DATE __________

APPROVED (EA-E2)____________________ DATE __________

	Department	date	Name	filename
Author	ZS-M-57	20.09.03	Frank	Disa.doc

Changes:

Version	date	comment
1.0	09/20/2003	First version

Table of Contents

1 FUNCTIONAL DESCRIPTION

The DISA causes a switchover between a long (torque setting, DISA on) and a short (power setting, DISA off) intake path.

With the switching DISA used with EVT, the switching point is determined by a lower speed limit K_DISA_N_EIN, an upper speed limit K_DISA_N_AUS and by the full load condition B_VL.

The DISA is in the on state if the full load condition is valid and the speed is in the range K_DISA_N_ON < n < K_DISA_N_OFF, otherwise the DISA is off.

The DISA is adjusted via an electric motor that is controlled by a PWM.

1.1 CONDITIONS OF DISA

The DISA has four different states:

disa_state state	
0	DISA off (performance position)
1	Adjust DISA from off to on
2	DISA on (torque setting)
3	Adjust DISA from on to off

In idle states 0 and 2, the DISA is controlled via a 20% PWM signal
corresponding polarity in order to prevent the DISA from adjusting itself due to vibrations.

During the switchover processes (disa_state 1 and 3) there is a control dependent on a characteristic (KL_DISA_TV) with a PWM signal between 100% and 20%.

1.2 INITIALIZATION

The initialization takes place in the disa_init function.
After the initialization, the DISA is controlled with a 20% PWM signal in the direction from, disa_state is set to zero.

The DISA is then in the off state.

1.3 SWITCHING THE DISA

DISA is switched over in the disa_10ms function.

The DISA is only switched over as long as the condition engine is running (B_ML) is true.

1.3.1 SWITCH ON

After initialization, the DISA is in the power state, ie disa_state = 0.

A switchover occurs when the following conditions apply:

- DISA in performance position: disa_state = 0
- Speed greater than K_DISA_N_EIN: n > K_DISA_N_EIN
- Speed lower than K_DISA_N_AUS: n < K_DISA_N_AUS
- Motor in Vollast: B_VL = 1

If all four conditions are true, disa_state = 1 is set.
As long as disa_state = 1, the function disa_ein() is called (10ms cycle).

The disa_ein() function outputs the appropriate direction bit for the correct polarity and a PWM signal.

The PWM duty cycle is determined by the applicable characteristic KL_DISA_TV, the input variable of the characteristic is the counter variable disa_cnt.
disa_cnt is incremented each time disa_ein() is called, so the characteristic curve is traversed.

First, a 100% pulse duty factor is output, which is then reduced to 20% to avoid jamming at the on position stop.

The last output duty cycle of 20% and the direction remain set until the next switching process.

If disa_cnt exceeds the value K_DISA_CNT_ENDE, the switching process is complete, disa_cnt is set to 0, disa_state to 2, the DISA is now in the torque position.

1.3.2 POWER OFF

The DISA is switched off when the following conditions apply:

- DISA in Momentenstellung: disa_state = 2
- one of the following three conditions:
  - on > K_DISA_N_AUS + K_DISA_HYST
  - on <K_DISA_N_EIN + K_DISA_HYST
  - o Condition full load B_VL is false

An applicable hysteresis K_DISA_HYST is added to the speed limits in order to avoid constant switching at the speed limits.

If the first and one of the following three conditions apply, disa_state is set to 3.
As long as disa_state = 3, the function disa_aus() is called.

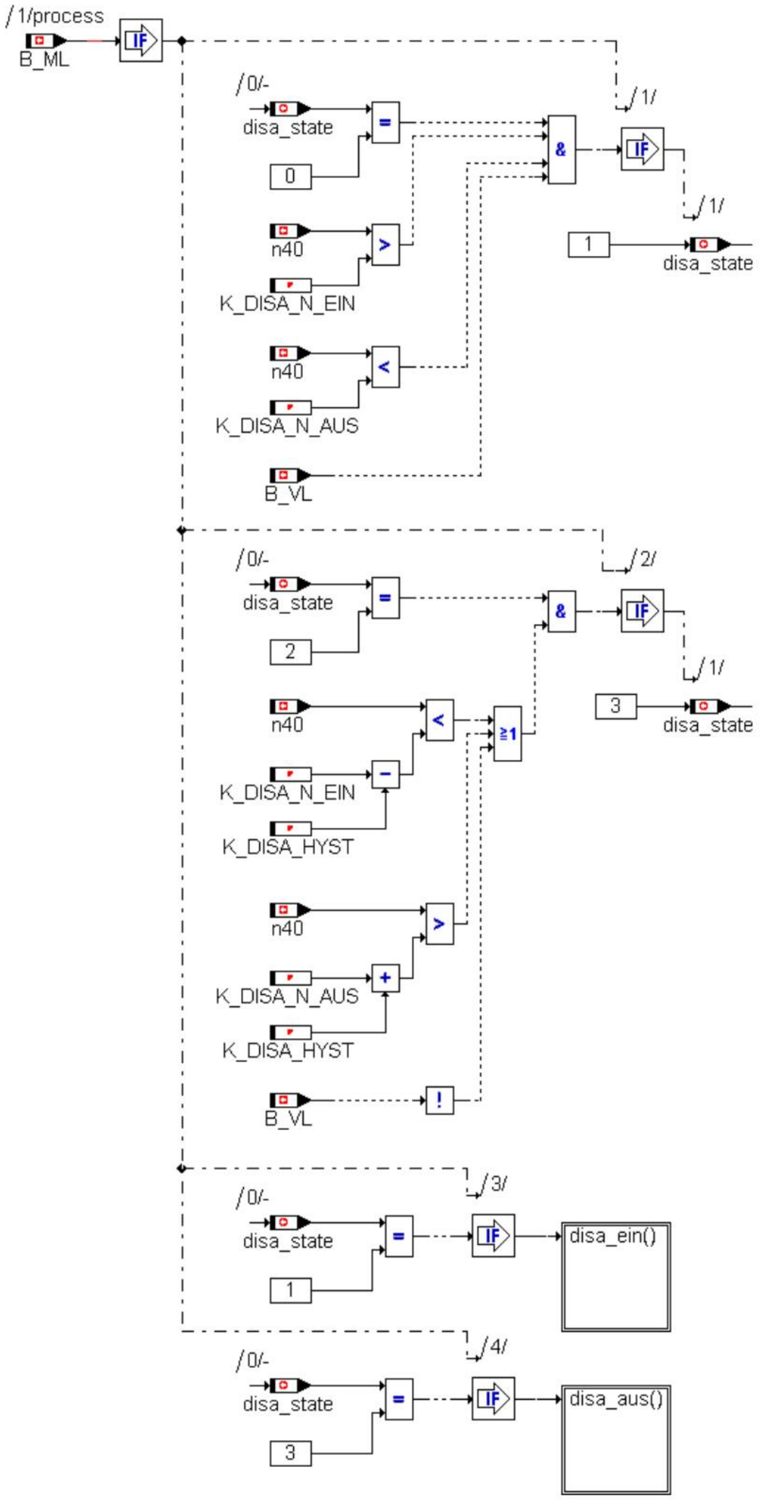
The direction bit is set in the opposite direction, the duty cycle is again calculated from the KL_DISA_TV characteristic.

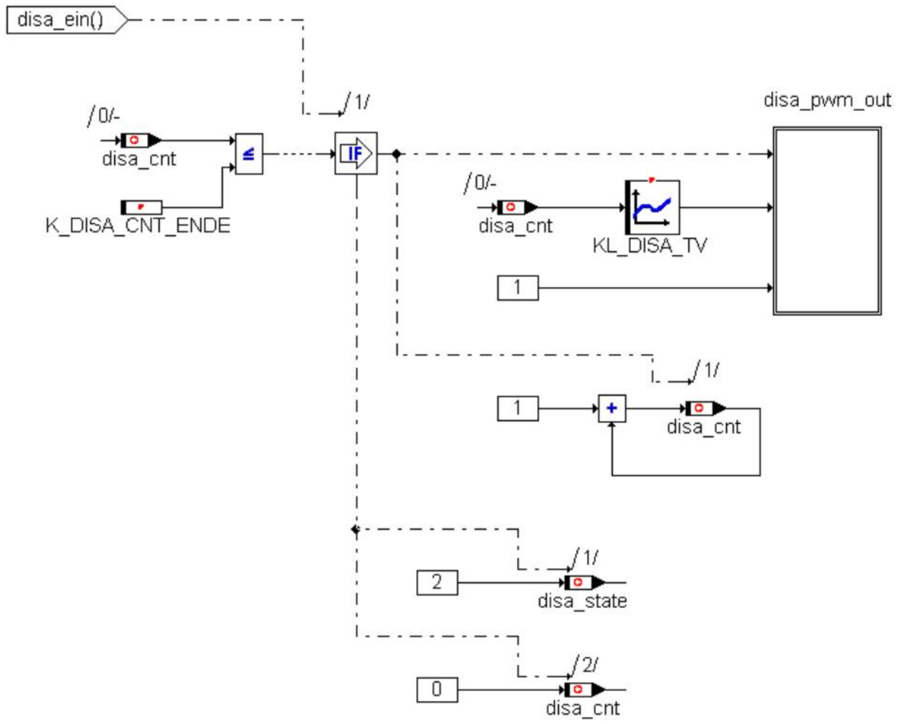
As soon as disa_cnt has exceeded the value K_DISA_CNT_ENDE and the characteristic has been traversed, disa_cnt and disa_state are set to zero, ie the DISA is now in the power setting, the switching process is complete.

1.4 REVERSAL OF DIRECTION

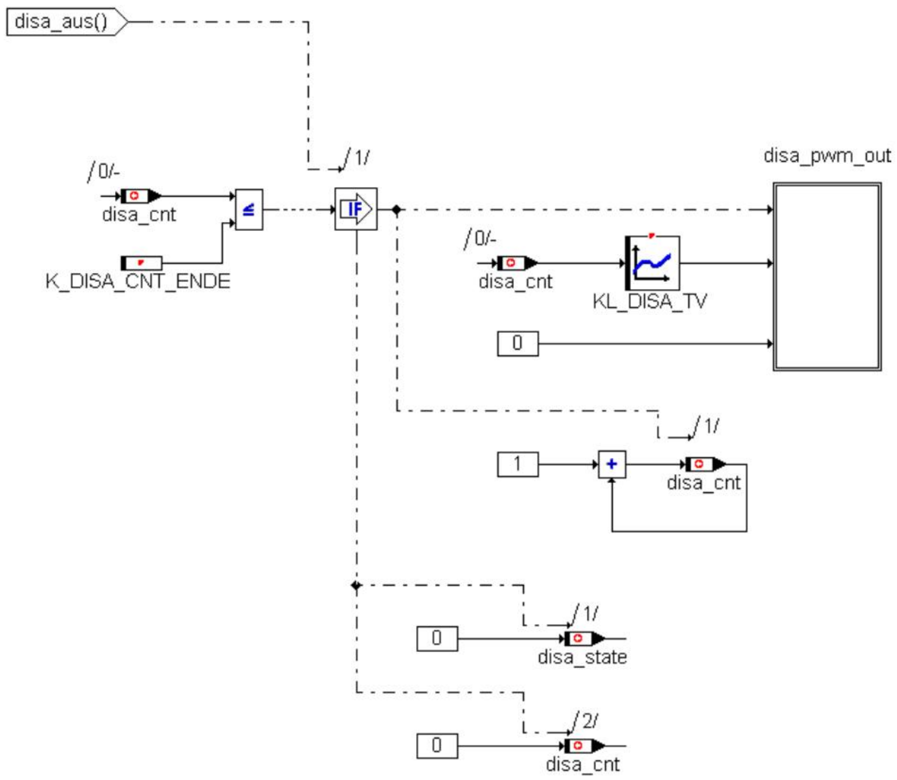
The switching direction of the DISA can be reversed with the constant K_DISA_DIR.
Since the direction bit of the hardware driver is only set in the event of a switching process, a switching process must be triggered after changing the constant K_DISA_DIR in order to recognize the change to take effect.

1.5 FUNCTIONAL DIAGRAM





disa_ein()
/1/
/0/-
disa_cnt
K_DISA_CNT_ENDE
disa_pwm_out
/0/-
disa_cnt
KL_DISA_TV
1
/1/
1
disa_cnt
/1/
2
disa_state
/2/
0
disa_cnt



2 DATA OF DISA

The function is calculated in the 10ms task.

Description of the variables:

disa_state	Operating state DISA	where

Description of the application data:

K_DISA_DIR	Reversal of direction	where
K_DISA_N_EIN	DISA lower speed limit	where
K_DISA_N_AUS	upper speed limit	where
K_DISA_HYST	Hysteresis value speed	where
KL_DISA_TV	Characteristic curve for duty cycle	ub / ub

MSS54

Modulbeschreibung EGAS

Table of contents: (automatically from chapter headings)

1. EGAS

The setpoint for the position controller of the EDK is determined in the EGAS module.

2. TARGET DETERMINATION

The setpoint is determined in the 10ms task. First, **egas_soll_bestimm()** is used to determine from which variable the setpoint is calculated. egas_soll_status is used to communicate the **variable** from which the setpoint is to be calculated. The final calculation of **egas_soll** takes place in **egas_soll_berech()**

instead of.

Since the setpoint is updated every 10 ms, but the actual value is recorded with each controller cycle, the result is a sawtooth curve and thus changing gradients of the control deviation. Gradually increasing the setpoint until the next update via **edk_soll_inc** results in a more homogeneous progression of the control deviation (see module description edk).

2.1. SETPOINT IN NORMAL OPERATION

egas_soll_stat = 1 or 2. The bit constant **CFG_M.EGAS** can be used to select whether the setpoint for the position controller is derived from the calculated setpoint torque of the motor or from the driver's request from the PWG.

CFG_M.EGAS = 0 => setpoint from PWG **(egas_soll** = **md_fw_rel)**
CFG_M.EGAS = 1 => setpoint from MM **(egas_soll** = **wdk_soll).**

After the ignition is off and **n** =0, the target value is taken from the specifications for adapting the upper stop.

In order to enable movement of the DK (e.g. diagnosis) when the adaptation is not active and the motor **(B_MS)** is stationary, there is a switchover to **md_fw_rel .**
If the condition **B_SKM_EDK_AUS** is met via the safety concept, a target value of 0 is specified.

2.2. SETPOINT VIA DIAGNOSTICS

egas_soll_stat = 3. The routine **edk_write(Parameter1, Parameter2)** can be called via diagnosis (DS2 protocol) . If parameter1 has the value 0, parameter2 is interpreted as a setpoint in % DK position, if parameter1 = 1, a pulse duty factor between 0% and 100% is transferred via parameter2. The desired value is only set when the engine is stationary **(B_MS)** and in diagnostic **mode (B_DIAG) .** In the background task, it is constantly checked whether these conditions are still met **(edk_write_undo)** and, if necessary, the control is switched off via diagnostics.

2.3. SETPOINT DURING ADAPTATION IN FOLLOW-UP

egas_soll_stat = 4. During the 100% adaptation in the run-on, **edk_soll_adapt** is taken as the setpoint.

2.4. TARGET VALUE DURING PRE DRIVE CHECK

egas_soll_stat = 5. During the pre-drive check after KL15 on, K_PDR_SOLL_EDK is **taken** as the setpoint.

Error! Not a valid shortcut.

3. CONSTANTS, CHARACTERISTICS AND VARIABLES

3.1. CONSTANTS

K_EGAS_UBMIN	Threshold for enabling 100% adaptation
K_EGAS_WDK_TAU	

3.2. CHARACTERISTICS

KF_EGAS_WDK	Conversion rf_soll to wdk
KF_EGAS_WDK_KH	Conversion rf_soll to wdk for cat heating
KF_EGAS_WDK_ENTDROSSELT -	

3.3. VARIABLES

egas_soll_status 1:	Status setpoint specification
2: 3: 4: 5: 6:	wdk_soll
	md_fw_rel
	edk_soll_diag
	edk_soll_adapt
	K_PDR_SOLL_EDK
	0 (for PDR)
egas_setpoint	Target EGAS
egas_actual	Actual EGAS
egas_ipk	Communication Master - Slave for EGAS
Bit 1:	Request master to slave, switch off H-bridge
Bit 2:	Slave to master confirmation, H-bridge switched off
Bit 3 to 7:	free
wdk_soll	Setpoint specification EGAS

MSS54

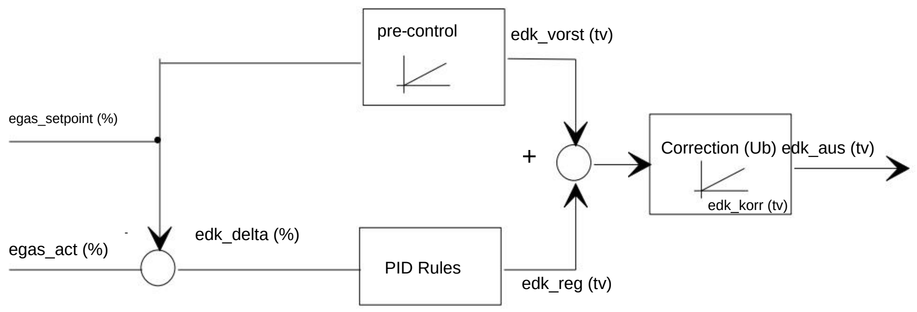
Modulbeschreibung
EDK

Table of Contents: (automatically from chapter headings)

1. OVERVIEW

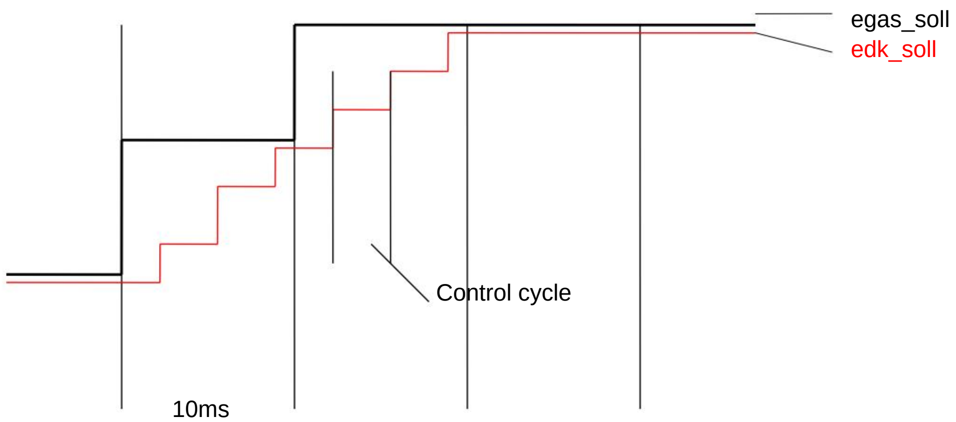
The throttle valve control consists of the following parts:

• Calculation of the pre-tax value
• Stock rules
• Correction of the output value via Ubatt
• Diagnose



2. TARGET DETERMINATION

The setpoint **egas_soll** is determined in the EGAS module every 10 ms.
An additional setpoint **(edk_soll)** is managed internally, which is brought up to the new setpoint in several steps depending on the controller cycle when there is a change in **egas_soll .**



This internally controlled setpoint avoids jumps in the control deviation **(edk_delta),** which would otherwise occur with every update of the setpoint.

3. ACTUATOR FEEDBACK DETECTION

The current DK potentiometer is used as the actual value **(egas_ist)** (see WDK module).

3.1. ADAPTION

3.1.1. ZERO POINT ADAPTATION

The zero point adaptation of the throttle valve takes place during the PredriveCheck (see modules PDR and WDK).

3.1.2. FULL LOAD ADAPTATION

The full load adaptation of the DK system takes place in the run-on if one of the following conditions is met:

- Virgin Control Unit
- Loss of adaptation data
- Call via DIS
- Error message about the security concept

Procedure:

After the ignition is off and n = 0, the following sequence is run through once:

- Approaching the setpoint to **K_EDK_A100_B1** (approx. 85%) via ramp **K_EDK_A100_INC**
- Waiting time **K_EDK_A100_WAIT1,** so that the setpoint can be adjusted safely.
- Further ramp-shaped increase of the setpoint by **K_EDK_A100_INC2** until the actual value can no longer follow ÿ permanent controller **deviation K_EDK_A100_DELTA**
- Waiting time K_EDK_A100_WAIT2 with checking whether the stop value remains stable (at further increase demand).
- Call of the **wdk_a100_adapt()** routine for adapting the DK • Drive
the EDK to zero via the ramp (set value with each cycle by **K_EDK_A100_DEC** to decrease).
- Exit control

4. CONTROL OF THE ACTUATOR

4.1. FEEDBACK

The servomotor must hold the desired throttle valve position against the throttle valve return springs. For this reason, a duty cycle is used as a feedforward control

Compensation of the spring force issued. This output value is calculated in the 10ms task from the **KL_EDK_VORST** characteristic .

edk_vorst = KL_EDK_VORST = f (egas_soll)

4.2. STOCK RULES

The position control runs as a PID controller according to the formula

Y = xp + xi + xd.

where the P part the I part the D part **xp = e * Kp xi = e * Ki + xit-1 xd = (et - et-2) * Kd** ist.

The control deviation **e** is calculated from the difference between the setpoint **(egas_soll)** and the actual value of the servomotor **egas_ist** (or **edk_soll** to **egas_ist) .**

The P component is determined separately for positive and negative control deviation from the characteristic curves **KL_EDK_PPOS** and **KL_EDK_PNEG .**

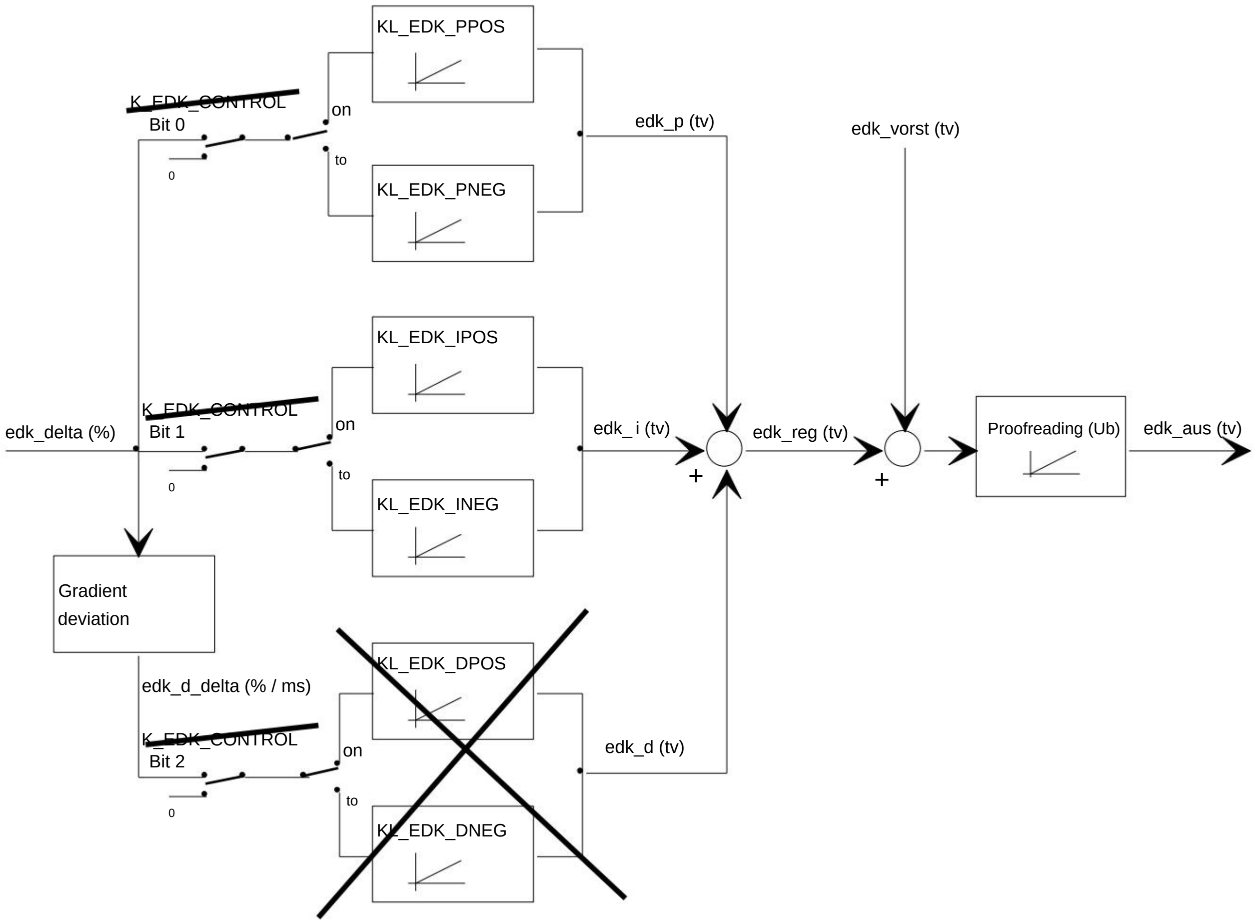
The I component is determined separately for positive and negative control deviation from the characteristic curves **KL_EDK_IPOS** and **KL_EDK_INEG .**
The I component is limited by **K_EDK_IBEGR .**
If the control deviation is greater than **± K_EDK_IDELTA** (eg if there is a jump), the I component is deleted.

The D component is determined separately for positive and negative gradients of the control deviation **(edk_d_grad)** from the constants **K_EDK_DPOS** and **K_EDK_DNEG .**

If the amount of the control deviation is greater than **± K_EDK_D_EIN_POS ,** the D component is switched off.

In the area around the zero point of the control deviation between **± K_EDK_D_NULL ,** the D component is switched off. If the target value **egas_soll** is greater than **K_EDK_D_ANSCHL,** switching off around the zero point of the control deviation is canceled in order to prevent the DK from oversteering into the mechanical stop.

The contribution of the D controller is limited to **± K_EDK_D_MAX .**



The output value determined in this way from **edk_reg** and **edk_vorst** is corrected using the characteristic **KL_EDK_KORR_U** via **Ub (edk_korr).**

The resulting drive duty cycle is as follows.

Edk_aus	=	(	edk_vorst +	pre-tax value
			edk_reg edk_korr	Adjustable value
		)		tension correction

4.2.1. CONTROL DISCONNECTION

If the engine is stationary, a setpoint of 0 and a WDK position <= **K_EDK_closed ,** the servomotor is switched off. When the engine is running, the pre-control value is output in order to slightly preload the kinematics in the direction of pull and thus reduce the backlash at the zero point.

4.3. DUTY CYCLE LIMITATION

The high time of the duty cycle edk_auss to be output is set via the variable **edk_ht_max** limited at the top.
The maximum achievable value is set via **K_EDK_MAX .**

edk_ht_max is calculated in the background task :
• In the event of an error in the EGAS system, it is limited to **K_EDK_SK_HT_MAX .**
• Limited to **K_EDK_A100_HT_MAX** during 100% adaptation

5. OWN DIAGNOSIS

5.1. DRIVER DIAGNOSTICS MC33186

Port E bit 4 can be used to determine via the diagnostics output of the H-bridge whether the bridge has switched off due to overload. This happens at the beginning of each controller cycle. At the end of the controller cycle, an attempt is made to switch the bridge back on if necessary. In the **edk_tr_diag_stat** variable, the hardware diagnostics routine **(edk_tr_diag())** is informed that the protective circuit of the bridge has responded.

When the bridge is switched off, bit 0 is set in **ed_edk_tr_stat .**
If the bridge is disabled, bit 1 is set in **ed_edk_tr_stat .**

If the bridge shuts itself down, an error is placed in **edk_hw_ed** (overtemp) bit 3.

error	**impact**	**measure**
overload of bridge	No activation of the servomotor	- Discard mistakes - Switch on the bridge again

5.2. SECURITY CONCEPT

See EGAS security concept module description.

6. DIAGNOSTICS VIA DS2

6.1. CONTROL OF THE DK VIA DS2

By calling **edk_write(edk_switch,edk_vorgabe)** the DK can be controlled via DS2.

The DK is only activated when the engine is stationary **(B_MS)** and in diagnostic **mode (B_DIAG).** If one of the two conditions is not met, there is no activation and response 4 (condition not met) is returned.
The **edk_switch** parameter distinguishes between whether the controller is given a setpoint (0) or whether the controller is controlled with a duty cycle (1). If another value is passed, there is no activation, the routine returns the result 2 (incorrect parameter).

A value from 0 to 200 is passed via the **edk_vorgabe** parameter. The setpoint of the position controller (0 to 100%) is determined from this value, resulting in a resolution of ½%.
Here, too, if another value is transferred, there is no activation and the feedback 2 (wrong parameter).

If all parameters are correct and all conditions are met, the answer is 0 (ok) and the DK are activated.

If the amount of the control deviation is less than **± K_EDK_DS2_DMAX ,** the specified setpoint is considered to have been adjusted.

6.2. OUTPUT OF SYSTEM VALUES VIA DS2

The following system parameters can be output via DS2:

adjustment time	**edk_ds2_t_stell**	Measures the time until the setpoint specified via DS2 is reached.
closing time	**edk_ds2_tschliess**	Time from switching off the controller to DK closed at the PDR.
maximum deviation	**edk_ds2_abw_umax** **edk_ds2_abw_omax**	Maximum control deviation that has occurred after reaching the setpoint set via DS2.
medium deviation	**edk_ds2_abw_mw**	Mean value of the amount of the control deviation after reaching the setpoint set via DS2.

7. CONSTANTS, CHARACTERISTICS AND VARIABLES

7.1. CONSTANTS

K_EDK_CONTROL	former connection of the individual controller components
Bit 5	Output pre-control value with switched off controller in LL (S54 only)
K_EDK_UBMIN	UB threshold for adaptation
K_EDK_CYCL	Controller cycle in ms
K_EDK_D_NULL	Switch-on limit D controller
K_EDK_D_MAX	maximum allowed D value
K_EDK_DPOS	K-Factor D-Rules positive
K_EDK_DNEG	K-Factor D-Rules negative
K_EDK_D_EIN_POS	Upper switch-on limit D controller positive
K_EDK_D_EIN_NEG	Upper switch-on limit D controller negative
K_EDK_D_ANSCHL	Deactivation of the zero limitation of the D controller
K_EDK_I_NULL	Freezing limit of the I controller
K_EDK_IBEGR	Limitation of the I share
K_EDK_IDELTA	Control deviation above which the I component is deleted
K_EDK_A100_DELTA	Deviation from which the 100% adaptation of the specified target value is considered to have been reached
K_EDK_A100_WAIT1	Waiting time for 100% adaptation to area 1
K_EDK_A100_WAIT2	Waiting time for 100% adaptation to area 2
K_EDK_A100_INC1	Step size for upward adaptation (0 to B1)
K_EDK_A100_INC2	Increment for upward adaptation (from B1)
K_EDK_A100_DEC3	Increment when closing flap after upward adaptation
K_EDK_A100_B1	Setpoint specification for the first adaptation step, upper stop
K_EDK_A100_VL_ANSCHL	Difference from the mechanical upper stop to 100% EDK position
K_EDK_T_SPERR	Minimum time without change after which the controllers in the LL turns off
K_EDK_CLOSED	Threshold below which the controller is switched off
K_EDK_ HT_MAX	Maximum allowed TV
K_EDK_A100_HT_MAX	Maximum TV during adaptation
K_EDK_SK_HT_MAX	Maximum TV with emergency program
K_EDK_AUS_HT_MAX	Maximum TV after switching the bridge back on
K_EDK_HT_MIN	Smallest possible TV value
K_EDK_HT_INC	Increment when increasing the TV in the event of an error
K_EDK_HT_TMOT	Limit temperature below the TV is limited

K_EDK_ HT_AUSZEIT Time to limit TV after error

K_EDK_ DS2_DMAX Threshold, target value of DS2 reached

K_EDK_ DS2_TSPERR Waiting time, measure after adjustment time

K_EDK_ DS2_TAU Filter constant for mean control deviation

7.2. CHARACTERISTICS

KL_EDK_VORST Pre-control value from DK setpoint

KL_EDK_PPOS P-factor of the position controller deviation greater than 0

KL_EDK_PNEG P-factor of the position controller deviation less than 0

KL_EDK_IPOS I factor of the position controller control deviation greater than 0

KL_EDK_INEG I factor of the position controller deviation less than 0

KL_EDK_KORR_U Correction of the duty cycle via Ubatt

7.3. VARIABLES

edk_soll Setpoint specification from torque manager or PWG

edk_soll_adapt Target value default adaptation routine

edk_delta Control deviation in % 16-bit value

edk_hw_ed Hardware H-bridge status byte

Bit 0: Error Maximum value exceeded
Bit 1: Error below minimum value
Bit 2...7: free

edk_status Statusbyte EDK

Bit 0: 1: rules according to PWG 0: rules according to MM
Bit 1: 1: Controller shutdown requested
Bit 2: 1: Controller shutdown is active
Bit 3: free
Bit 4: Adaptation value a0 from EEPROM lost
Bit 5: Adaptation value a100 from EEPROM lost
Bit 6: 1: Adaptation active
Bit 7: 1: EDK adaptation has taken place

edk_tr_diag_stat Status byte driver diagnosis H-bridge

Bit 0: 1: SF error detected by position controller, jumper disabled
Bit 1: 1: Bridge must not be enabled
Bit 2...7: free

edk_lr_i Increment of the I controller from the characteristic curve via deviation

edk_p P component of the output value

edk_i I component of the output value

edk_d	D component of the output value
edk_reg	Adjustable value of PWM hightime
edk_vorst	Pre-control value of the PWM high time
edk_aus	Output value of the PWM high time
edk_korr_fak	Correction factor from characteristic over Ub
edk_korr	Input tax value corrected via Ubatt
edk_master_reset	Triggering a reset on the master by writing to it variables
edk_d_grad	Gradient of the control deviation
edk_soll _diag	Setpoint specification via diagnosis
edk_soll_inc	Increase from edk_soll to egas_soll (set point adjustment to cycle time)
edk_delta2	Control deviation edk_soll - edk_ist (setpoint adjustment to cycle time)
edk_d_grad2	Gradient of the control deviation (set point adjustment to cycle time)
edk_ht_max	Duty cycle limitation
edk_ds2_tstell	Adjustment time according to setpoint via DS2
edk_ds2_tschliess	Switch off closing time via spring after actuator
edk_ds2_abw_umax	Maximum control deviation below when controlling DS2
edk_ds2_abw_omax	Maximum control deviation above when controlling DS2
edk_ds2_abw_mw	Mean control deviation below when controlling DS2
edk_ds2_sollw_alt	last setpoint via DS2
edk_ds2_status	Status byte activation via DS2
edk_ds2_adapt_stat	Status byte control adaptation via DS2

MSS54

Modulbeschreibung PDR

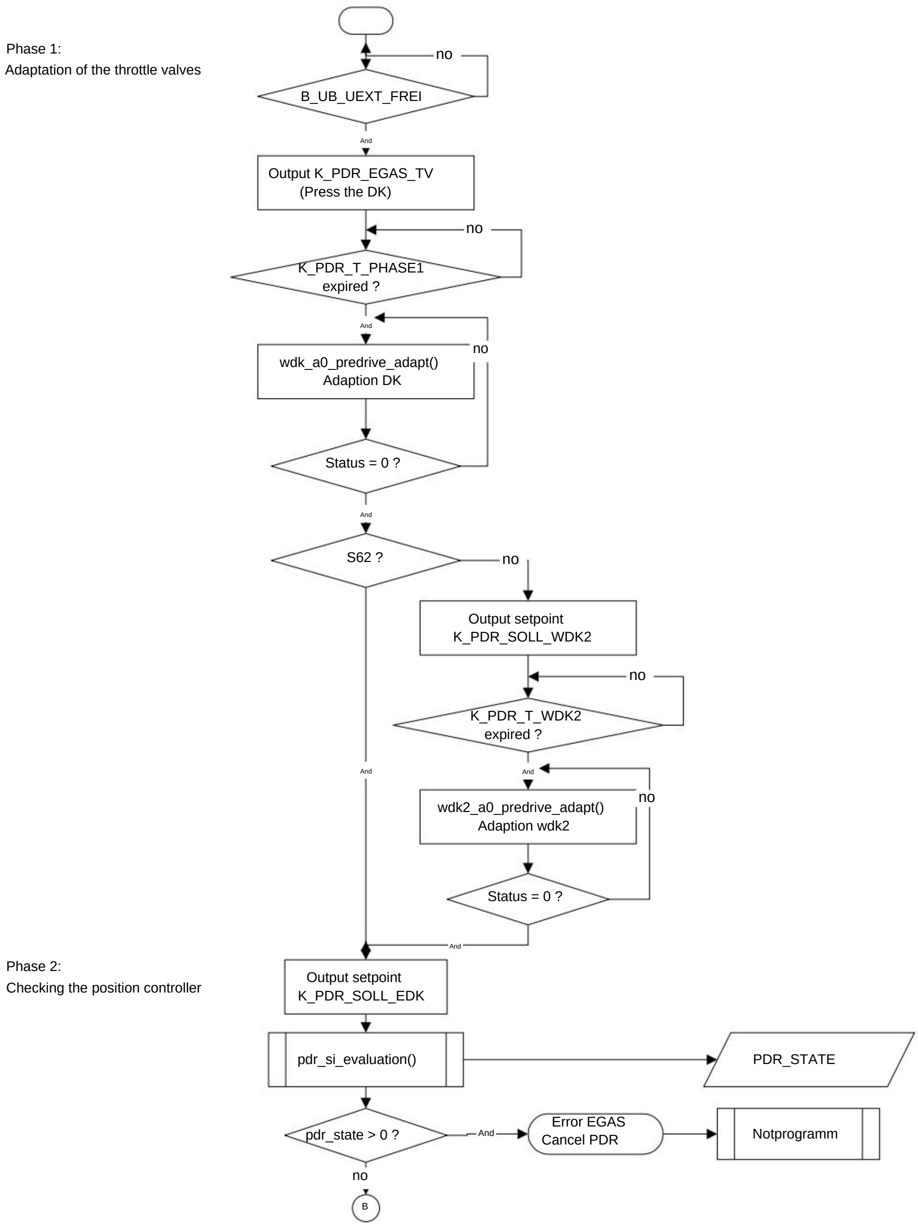
	Department	date	Name	Filename
editor	EE-221	04.12.2003		3.03

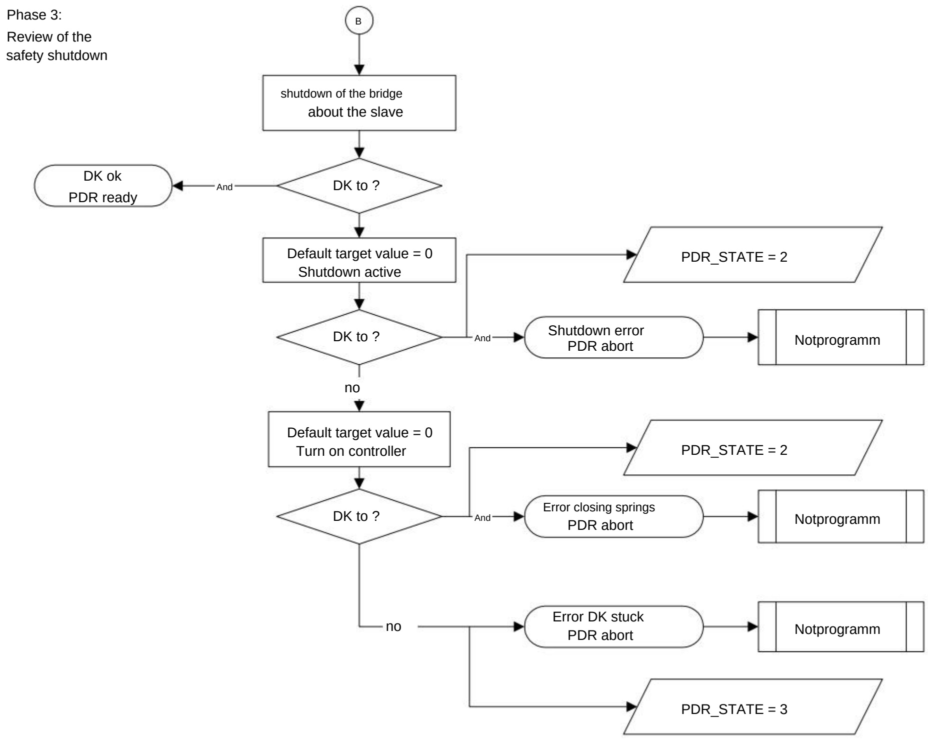
Table of contents: (automatically from chapter headings)

1. PRE DRIVE CHECK

After Kl15 on, a predrive check runs on the master.

pdr_m is called once after KL15 by task_pdr_m when the engine **is stopped .** During the predrive check, the zero point of the throttle valve is adapted **(pdr_phase** = 1), the function of the position controller is checked **(pdr_phase** = 2) and then the safety shutdown of the slave is tested **(pdr_phase** = 3). Phase 1 is run through after every start and cannot be interrupted, phase 2 and phase 3 are interrupted at engine speed. During phase 1, ignition and injection are blocked.





1.1. ZERO POINT ADAPTATION OF THE DK (PHASE 1)

The zero point adaptation differs for motor types S62 (V-8, 2 DK potentiometers) and S54 (R-6, one DK potentiometer and one potentiometer integrated in the controller).

- Squeeze the EDK with duty cycle **K_PDR_EGAS_TV** with the position controller switched off for **K_PDR_T_START Runs .**
- Remain with pressed DK for **K_PDR_T_PHASE1** runs.

B_CFG_S62 fulfilled (S62):

- Call of **wdk_a0_predrive_adapt** to adapt the DK (wdk1 and wdk2 on the S62).

B_CFG_S62 not fulfilled (S54):

- Call **wdk_a0_predrive_adapt** to adapt wdk1 via **wdk_a0_predrive_adapt**
- Setting the setpoint **K_PDR_SOLL_WDK2**
- Remain with the set target value for **K_PDR_T_WDK2** runs.
- Adaptation of the wdk2 by calling **wdk2_a0_predrive_adapt**

1.2. CHECKING THE POSITION CONTROLLER (PHASE 2)

- Switch on the position controller, setpoint is **K_PDR_SOLL_EDK.**
- Remaining with adjusted target value for **K_PDR_T_PHASE1** runs.

- Calling **edksi_query()** to evaluate the result of the check.

The control deviation of the position controller is checked via the setpoint/actual value comparison. The result will be in **pdr_state** laid down.

1.3. CHECKING SLAVE SAFETY SHUTDOWN (PHASE 3)

- When the setpoint is regulated, the slave is requested via Bit0 in **egas_ipk** to switch off the bridge.
- Wait until **dkm_word is** less than **K_PDR_DKM0** or timeout (counter of **K_PDR_T_PHASE3** expired).
  - If DK is not closed, specify position controller setpoint 0.
    - If DK is now closed, the safety shutdown of the slave does not work. => error
    - If DK is still open, end the slave shutdown.
      - If DK is now closed, the DK can be pulled mechanically via the spring packs not. => error
      - If DK is still open, the DK is stuck. => error.
- End of the EGAS check, switch on the bridge from the slave, position controller on, setpoint off normal operation.

After completion of the PDR, Bit7 is set in **pdr_status .**

If the DK does not close within the counting time **(K_PDR_T_PHASE3)** when the controller is switched off , Bit3 in **pdr_status** set.

Detected errors are entered into the error memory via **pdr_ed .**
If necessary, various emergency programs are branched via **pdr_status .**

error	**impact**	**measure**
Timeout DK 0% Adaption		Notprogramm 1
Timeout Stock rules	EGAS system cannot be operated reliably	Emergency program 1,2 or 4 (depending on result of edksi_query)
Controller shutdown Slave does not work.	Security concept loses possibility of intervention	Notprogramm 2
Closing springs defective	After a safety shutdown, the flaps do not close.	File DK error, emergency program 2
Clap clamps		Switch off DK, emergency program 4

1.4. MORE TESTS

If necessary, further tests can be appended to State 13.

1.5. CONTROL VIA DIAGNOSIS

When the motor is stationary, the DS2 interface can be used by calling **pdr_write()** the predrive check can be triggered.

2. CONSTANTS, CHARACTERISTICS AND VARIABLES

2.1. CONSTANTS

K_PDR_T_START	Press the expiration time DK
K_PDR_T_PHASE1	Press expiry time DK and adaption wdk
K_PDR_T_PHASE2	Adjust expiration time DK
K_PDR_T_PHASE3	Expiration time accrues to DK
K_PDR_A0_TIMEOUT	Timeout DK Zero point adaptation
K_PDR_SOLL_EDK	Setpoint specification position controller
K_PDR_EGAS_TV	Setpoint duty cycle
K_PDR_EDK_DMAX	allowed deviation for position controller
K_PDR_DKM0	allowed deviation for position controller off
K_PDR_T_WDK2	Adjust expiration time DK for adapting wdk2 (S54 only)
K_PDR_SOLL_WDK2	Setpoint specification position controller for adapting wdk2 (S54 only)

2.2. CHARACTERISTICS

KL_PDR_???	None yet

2.3. VARIABLES

pdr_status Statusvariable

Bit 0:	PDR active
Bit 1:	Setpoint specification from PDR
Bit 2:	Position controller does not reach setpoint
Bit 3:	DK does not close (time out switching off from the slave)
Bit 4:	Time out Adaption DK
Bit 5:	Error switching off slave -> setpoint specification = 0
Bit 6:	free
Bit 7:	Run through PDR and exit

pdr_ed Diagnosestatus PreDRive Check

Bit 0:	Clap Clamps
Bit 1:	timeout shutdown by the slave
Bit 2:	timeout EGAS Inventory Rules
Bit 3:	Closing springs defective
Bit 4:	
Bit 5:	Error counter greater than 0
Bit 6:	Error entered in error memory
Bit 7:	

pdr_phase Progress indicator PDR

0:	Waiting for Uext ok
1:	Adaption DK
2:	Test Stock Rules
3:	Test Notabschaltung Slave
4:	PDR ready

pdr_m_state Statusvariable PDR

0:	PDR ok
1:	Adaption DK failed
2:	DK unplausible
3:	(there is not any)
4:	Actual value too large
5:	PDR unplausibel

pdr_abl_count state counter

MSS54

Modulbeschreibung

Electric throttle

Target/actual comparison

1 Target/actual comparison EGAS position

Comparing the target position of the throttle flaps with their actual position is one of the most important monitoring functions in the Egas safety concept. This allows the following errors to be identified:

- Processor modules
  - CTM module (processor): Uncontrolled duty cycle for servomotor
  - Processor port C: Incorrect direction of rotation of the servomotor
  - Processor port C: Missing release servomotor function computer
  - Processor port C: Missing release of safety computer servomotor

- H-bridge actuator
  - H-bridge defect
  - Over-temperature shutdown
  - H-bridge current limit
  - H-bridge overcurrent shutdown

- Wiring servomotor
  - Line break
  - Short-circuit to ground, Ub, or between the lines

- Actuator
  - Electrical defect
  - Mechanical damage
  - Transmission damage

- DK-Kinematics
  - Mechanical damage
  - Foreign body interventions

- Throttle valves
  - party-clamping flap
  - Throttle valve adaptation
  - Shift of the zero point
  - Displacement of the anchor point

The target/actual comparison does not take place in the following operating states:

- If the position detection has already reliably detected errors (B_WDK_POTIUNPLAUSIBEL), since measures are already being taken on this

- or with KL15 off and n = 0

- or when the engine is not running and
  - Follow-up adaptation 100% position is running (B_EDK_ADAPT)
  - or nothing has yet been adapted in the predrive (pdr_phase == 0)
  - or the zero point adaptation is currently running in the predrive (pdr_phase == 1)
  - or shutdown paths are tested in the predrive (pdr_phase == 3)

1.1 Case 1: The throttle valves are to be opened above a threshold that But flaps remain closed:

Reasons:

- Processor module defective
- H-bridge defective or briefly switched off
- Safety shutdown activated
- Actuator wiring
- Servomotor defective
- DK-Kinematics defective

Error detection:

The actual position is close to "0" although the target position has already exceeded a threshold: **egas_soll** > **K_EDKSI_POS_ZU** + **K_EDKSI_HYS_ZU**

and

egas_ist < **K_EDKSI_POS_ZU**

for longer than **K_EDKSI_T_BL_ZU**

Reaction:

• Change to Egas emergency program stage 2 - driving over idle controller • Error memory entry

Evaluation:

The throttle flaps remain closed or are closed independently by the spring packs without the control unit being able to influence this.
The reduction in torque when closing the flaps cannot be influenced either (critical condition for case 1). If the flaps are closed, it is possible to continue driving in the emergency program without any problems if it is ensured that the flaps can no longer open.

1.2 Case 2: The throttle valves should be closed, but remain one Gap open:

Reasons:

- Throttle valve is jammed or extremely difficult to move
- Slight twisting of the control potentiometer of the throttle valve system
- Wrong zero point adaptation

Error detection:

If the setpoint is "0", the flap is slightly open, but still below a critical threshold: **egas_soll = 0**

and

K_EDKSI_POS_ZU < **egas_ist** <= **K_EDKSI_HYS_BL_AUF**

for longer than **K_EDKSI_T_SPALT**

Reaction:

- No Egas emergency program, maintaining the current operating level
- Error memory entry

Evaluation:

Since the throttle flaps remain open a gap despite being pressed shut via the servomotor, this indicates a problem in the DK system, which justifies a fault memory entry. However , the **K_EDKSI_HYS_BL_AUF limit** is dimensioned in such a way that it is not yet regarded as safety-critical for driving operation and therefore no change to an Egas emergency program has to take place.

1.3 Case 3: The throttle flaps should be opened, but the flaps react but do not reach the target value:

Reasons:

• H-bridge briefly switched off
• Sluggish DK system • Throttle stuck below the target position
• Undervoltage
• Motor too weak
• Transmission damage

Error detection:

The error exceeds a limit, the actual position is somewhat open, but not yet close to 100%:

ub> K_ED_UBMIN
and
egas_soll - **egas_ist** > **K_EDKSI_HYS_U_SOLL**
and
K_EDKSI_POS_ZU < **egas_ist** <= **K_EDKSI_POS_N_GANZ**

for longer than **K_EDKSI_T_U_SOLL**

Reaction:

• Change to Egas emergency program stage 2 - driving over idle controller • Error memory entry

Evaluation:

Since the reliability of the Egas system can no longer be guaranteed, the flaps are closed in a targeted manner and the control is then deactivated. If the flaps are jammed, a change to stage 5 is possible as soon as the setpoint is below the actual value.

1.4 Case 4: At full load, the throttle valves do not open fully:

Reasons:

• Flaps at the full load stop => incorrect adaptation
• Mechanical defect or foreign body limits the adjustment range
• Motor too weak
• Flaps stiff (extreme cold)
• Undervoltage

Error detection:

The error exceeds a limit and the actual position is close to 100%:

ub> K_EDKSI_UB_N_GANZ
and
tmot> K_EKDSI_TMOT_N_GANZ
and
egas_soll - **egas_ist** > **K_EDKSI_HYS_N_GANZ**
and
egas_act > **K_EDKSI_POS_N_GANZ**

for longer than **K_EDKSI_T_N_GANZ**

Reaction:

- No Egas emergency program, maintaining the current operating level
- Limitation of the Egas target value to the achievable actual position (servomotor protection)
- Start of a new VL adaptation in the run-on
- Error memory entry

Evaluation:

This case only results in a loss of performance in the full load range and is therefore not critical to safety. However, measures must be taken to protect the servomotor.

1.5 Case 5: The throttle valves are stuck in the open state:

Reasons:

- Defective processor module - 100% control, wrong direction of rotation
- H-bridge alloyed
- Short circuit in servo motor wiring
- stiff DK system
- Throttle stuck above target position

Error detection:

The actual position is significantly larger than the target position:

egas_actual - egas_soll > **K_EDKSI_HYS_BL_AUF**

a) for longer than **K_EDKSI_T_BL_AUF_R**
b) for longer than **K_EDKSI_T_BL_AUF_F**

Reaction:

a)

- Torque limitation via ignition angle and injection suppressions
  (Set condition **B_EDKSI_MD_RED)**

additionally off

b)

- Change to Egas emergency program stage 4 - driving with open throttle valves • Error memory entry

Evaluation:

In this case, the engine generates more power than the driver desires and unwanted vehicle acceleration can occur. This requires a quick response to this condition. But the controller has it

Opportunities to throttle the engine power to a range that the driver desires via ignition angle interventions and cylinder suppression.

2 Status/fault memory entries:

If a target/actual comparison error is detected, the type of error is classified according to the following priority in **edksi_status** entered and marked in the error memory with the following error type:

Priority 1st	**edksi_state**	**error type**
2nd 3rd 4th	STAY AWAKE	SH_TO_UB
5th	STAY_TO	SH_TO_GND
	UNDER_TARGET	UNPLAUSIBEL
	NOT_FULLY_OPEN	IMPLAUSIBLE
	GAP_OPEN	OPENLOAD

3 Applicable variables and process variables

In this section, all applicable constants, characteristic curves and maps are given in tabular form. The process variables that can be observed via the MCS are also specified.

3.1 Process Variables

Name	**description**
edksi_status	Status target/actual comparison
edksi_md_red	DPR: premature torque reduction, without error memory and emergency program
edksi_ed	ED error variable
edksi_t_bl_zu	Timer flaps remain closed
edksi_t_spalt	Timer flaps remain open
edksi_t_u_soll	Timer flaps remain below target
edksi_t_n_ganz	Timer flaps do not open fully
edksi_t_bl_auf	Timer flaps stay open

3.2 Constants

constant	**importance**	**minimum value**	**maximum value**
K_EDKSI_T_BL_ZU	Flap time not opening	Worst case response time	No surprise effect for the driver if the flaps open with a delay
K_EDKSI_T_SPALT	Error time for not completely closing flap	Worst case closing time	Error detection inside acceptable time.
K_EDKSI_T_U_SOLL	error time for stay behind under setpoint	Worst case rule time for the deviation **K_EDKSI_HYS_U_SOLL**	Error detection inside acceptable time.
K_EDKSI_T_N_GANZ	Error time for not reaching the full load position	Worst case control time for opening 0=>100%	The time should be measured in such a way that the error can also be detected during shorter full-load phases, but at least as long as **K_EDKSI_T_U_SOLL.**
K_EDKSI_T_BL_AUF_R	**Error time** for reduction with torque reduction to too wide open Succeed	Worst case rule time for the deviation **K_EDKSI_HYS_BL_AUF**	Critical time at which there is just no danger with hanging flaps without torque reduction.
K_EDKSI_T_BL_AUF_F	**Error time** for transition to emergency program when closed	Worst case rule time for the deviation	Time after which a defect in the system must be assumed

constant	Meaning of	minimum value	maximum value
	wide open flaps	**K_EDKSI_HYS_BL_AUF**	
K_EDKSI_T_PDR	time for Deviations that occur when queried by Predrive, from the than errors are reported	WorstCase Regelzeit	must be shorter than the waiting time **K_PDR_T_PHASE2** im Predrive Check
K_EDKSI_POS_ZU	Position below the flap is considered to be closed	zero-point fluctuation range Zero point adaptation error.??	A jump from 0 to this value must not cause a critical situation
K_EDKSI_HYS_ZU	minimal setpoint increase for error detection "Flaps stay closed"	Worst case deviation within from **K_EDKSI_T_BL_ZU**	**K_EDKSI_HYS_ZU** + **K_EDKSI_POS_ZU** must be large enough for the flap to have enough control deviation for reliable response.
K_EDKSI_HYS_U_SOLL	Max. permissible lag of the flaps behind the setpoint	max. possible control deviation under worst-case conditions within **K_EDKSI_T_U_SOLL** from	Error detection still possible
K_EDKSI_HYS_N_GANZ	Minimum setpoint increase for error detection "Won't open all the way"	max. possible control deviation under worst-case conditions within **K_EDKSI_T_N_GANZ** from	Reliable detection of stop in the opening direction
K_EDKSI_POS_N_GANZ	Actual position of the flap to the Distinction "Below target" and "Not quite up"	Position that a faulty drive can no longer reach.	must represent a range from which a mechanical attack is suspected
K_EDKSI_HYS_BL_AUF	Minimum Actual value increase for error detection "flaps stay open"	Maximum control deviation below worst Case Conditions inside **K_EDKSI_T_BL_AUF_R**	Even less critical moments excess through this Control deviation, especially when the target value is close to 0 (at target=0 target slam moment surveillance)
K_EDKSI_UB_N_GANZ	Minimum voltage for Diagnosis "Not quite up"	Voltage from which the servomotor reaches 100%	Minimum voltage when running Motor
K_EDKSI_TMOT_N_GAN WITH	Minimum tmot for Diagnosis "Not quite up"	Engine temperature from which the Actuator that reaches 100%	Significantly below engine operating temperature

MSS54

Modulbeschreibung

working version

Ega's security concept

(provisionally)

	Department	date	Name	Filename
Processor	EE-221	5.12.03		3.05

.1201.04.20
13 12:30:00

1. **GENERAL**

2. HARDWARE **SECURITY CONCEPT**

The design of the control unit hardware was specially geared to the needs of a safety-critical Egas system. It has a number of features that ensure uninterrupted monitoring and secure fail-safe of the Egas system.

Brief overview of the design features

Two processor system with two equivalent, powerful 32-bit processors

except for the power supply, completely independent processors with their own Clock supply and program/data storage

Use of a voltage regulator with reset triggering in the event of undervoltage

Division of the functionality based on the premise that each processor has the ability to independently intervene in the torque output of the engine.

Function calculator:	Egas system, idle actuator system, ignition
Security Calculator:	Injection including speed limitation

Use of an H-bridge to control the Egas servomotor with two switch-off paths, with each switch-off path being controlled by a processor.

Redundant allocation of the brake light switch to both processors

Redundant distribution of the two analog signals from pedal position sensor 1 and throttle valve sensor 1 to both processors

Use of a pedal value transmitter with two potentiometers PWG1 and PWG2 with independent power supply and different characteristics.

Installation of two throttle position sensors with independent voltage supply and crossed characteristic.

Dual 5V sensor supply. Reading back the supply voltage within the DME

Connection of the reset line of a processor to a port pin of the other processor. Port pin optionally configurable as interrupt input or output

3. **SECURITY CONCEPT SOFTWARE**

In order to be able to ensure safe Egas operation, a number of software modules are implemented in the MSS54, which detect and correct all possible SG-external (sensors, actuators, wiring harness) as well as SG-internal (processor, memory, driver, power supply) errors should transfer the system to a fail-save state.

The monitoring modules can be divided into three levels:

Level 1:	Monitoring of the SG periphery (sensors or actuators)
Level 2:	Monitoring of the control circuits, setpoint specifications Plausibility check of mutually redundant information
Level 3:	Monitoring of the control unit hardware and the proper program flow

The level 3 monitoring modules are implemented on both processors and run independently of one another, so that a computer unit failure does not pose a risk, since the monitoring module running in parallel is still working properly.

Brief overview of the monitoring modules:

sensors:

• Sensor supply Uext • Pedal value pwg • Throttle position wdk: • HFM load signal ml • Brake light switch system

area monitoring
Area monitoring, channel comparison
Area monitoring, channel comparison
area monitoring
Channel comparison

Actor:

• Idle speed controller • Egas servomotor

electrical driver diagnostics
electrical driver diagnostics

Comparison tests: • Target/actual comparison of the throttle valve position
• Plausibility check of the driver's desired torque in relation to the actual engine torque
• Plausibility check of load signal for throttle valve position (only if one fails DK-Potis)

Area monitors:
• Plausibility check of the torque calculation including torque-increasing interventions
• Monitoring DK position with zero torque specification
• Monitoring FGR shutdown when the brake is actuated

Interface monitoring:
• CAN interface - bus error, telegram timeout
• DSC interventions - Verification of signal redundancy
• MFL interface - timeout, telegram format, key coding

Monitoring SG hardware:
• QADC : Result comparison function and backup calculator
• Memory Tests
• CPU monitor test tasks

Testabläufe / Systemtests
• Pre Drive Check Egas-System
• Programmablaufkontrolle
• Reset monitoring
• Communication monitoring function / backup computer

4. PWG-NOTLAUFPROGRAMME

4.1. LEVEL A - PWG EMERGENCY DRIVING WITH A PWG SENSOR

Identification of the emergency program:
Unequivocally detected failure of a pedal sensor and thus loss of redundancy.

Prerequisite for PWG operation in level A:
Remaining pedal position sensor is plausible.
Brake switch system is error free
no internal control unit errors

Emergency program:
- Switching the pedal value progression curves to an emergency running progression curve.
- Limitation of the positive PWG dynamics through emergency operation filtering of the setpoint - slow Up filtering + fast down filtering
- Safety shutdown via brake light switch
  as soon as the brake is applied, a pedal value of zero is output. A new pedal value not equal to zero is only accepted again if the remaining pedal value transmitter has meanwhile returned to the value zero.

Remarks:
- Cruise control operation is still possible without restrictions.

4.2. STUFE B - PWG-NOTFAHREN OHNE PWG-SENSOR

Identification of the emergency program:
Failure of both pedal sensors - a driver's request can no longer be detected.

Prerequisite for PWG operation in level A:
no sg-internal errors

Emergency program:
- The driver's request is always zero
- Driving at idle speed

Remarks:
- Cruise control operation is still possible without restrictions if the minimum speed for the FGR operation can be achieved.

5. EGAS-NOTLAUFPROGRAMME

5.1. STAGE 1 - DK EMERGENCY DRIVING WITH A DK SENSOR

Identification of the emergency program:
Unequivocally detected failure of a throttle valve sensor and thus loss of redundancy, or mutually implausible DK values of the two sensors without having recognized the defective sensor. In this case, the Egas position control uses the larger and therefore less critical value as the actual position of the throttle valve until the faulty encoder can be detected via the HFM plausibility check.

Requirements for DK operation in level 1:
Remaining throttle position sensor is plausible.
HFM works flawlessly.
Plausibility check of remaining DK value for HFM load signal OK
no internal control unit errors

Emergency program:
• Limitation of duty cycle for Egas servomotor - limitation of engine dynamics
• Limitation of the maximum engine torque •
Plausibility check of remaining potentiometer via HFM load signal
• Blocking of the internal filling-increasing interventions such as cat heating, torque reserve
• Limiting the vehicle acceleration • Limiting the
maximum speed

Remarks:

5.2. STAGE 2 - EMERGENCY DRIVING VIA IDLE ADJUSTMENT SYSTEM

Identification of the emergency program:
The target position of the throttle valve can no longer be reliably adjusted because
• the actual position can no longer be detected due to a double error (DK1, DK2, HFM) or failure of the sensor supply • the actuator (driver, cable, servomotor, DK mechanism) has failed
• there is a problem in the throttle valve kinematics

Requirements for emergency operation in stage 2:
at least one load signal available (HFM or a reliable throttle valve position)
no internal control unit errors

Emergency program:
• Switching off the servomotor control and monitoring whether the throttle valves are closed.
• Limitation of the maximum engine torque •
Blocking of internal filling-increasing interventions such as cat heating, torque reserve
• Reducing the speed limit
• Limiting the vehicle acceleration • Limiting the
maximum speed

Remarks:

5.3. STAGE 3 - EMERGENCY DRIVING VIA IDLE ADJUSTMENT SYSTEM WITH OPEN THROTTLES

Identification of the emergency program:
The air supply to the engine can no longer be controlled directly, for example because the throttle flaps are stuck in an open position. The torque desired by the driver or the engine speed must therefore be reduced to a desired level via ignition and injection.

In this mode of operation, it can generally be assumed that there is a defect in the control of the throttle flaps, but that the actual positions can still be detected.

Requirements for emergency operation in stage 3:
no SG internal error

Emergency program:
• Switching off the servomotor control.
• Limitation of the maximum engine torque •
Enabling the ignition angle interventions of the torque manager (intervention becomes active when the actual engine torque is above the desired torque).
• Activation of the injection suppression torque manager (intervention becomes active when engine Actual torque is above the driver's desired torque + max. permitted delta)
• Blocking of the internal filling-increasing interventions such as cat heating, torque reserve
• Reducing the speed limit
• Limiting the vehicle acceleration • Limiting the
maximum speed

Remarks:
This emergency program represents the "worst case" in Egas operation. The engine generates more torque than the driver desires and the vehicle could unintentionally accelerate. However, since switching off the engine is also considered to be extremely safety-critical, this emergency program is intended to maintain a highly restricted but still controllable engine operation.

5.4. STAGE 4 - EMERGENCY DRIVING VIA IDLE CONTROL SYSTEM DUE TO CONTROL UNIT INTERNAL FAULT

Identification of the emergency program:
One of the control unit's monitoring functions has detected an error within the DME, which means that proper processing of the program can no longer be reliably guaranteed. Since the effects of the error cannot be foreseen in this case, a number of parallel and independent measures are taken to ensure that the vehicle cannot unintentionally accelerate as a result of this error.

Requirements for emergency operation in level 4:

Emergency program:
• Switching off the servomotor control.
• Limitation of the maximum engine torque •
Activation of the torque manager ignition angle interventions •
Activation of the injection suppressions torque manager
• Blocking of the internal filling-increasing interventions such as cat heating, torque reserve
• Reducing the speed limit
• Limiting the vehicle acceleration • Limiting the
maximum speed

Remarks:
Due to the structure of the DME as a two-processor system and the distribution of air supply and fuel supply to one processor each, the DME is able to guarantee safe emergency operation even in the event of a serious internal problem. Each processor is able to control the engine torque generated, regardless of the functionality of the other processor.

However, the drivability of the emergency program depends very much on the extent to which the functions required for engine operation, such as load detection, ignition, injection, etc., can still run without errors.

6. TRANSITIONS **TO FLAT RUN PROGRAMS**

As part of the Egas safety concept, there is a particular focus on the transitions to the corresponding emergency programs. Because while an emergency program appears manageable with suitable countermeasures, this always depends on the current driving situation during the transitions. However, this is usually not known to the controller.

Switching off the engine or a sudden power reduction of the engine is considered to be particularly critical, since this may provoke a driving condition that is critical to safety. Key points are: overtaking in oncoming traffic, driving at the limit with abrupt load change reactions, loss of steering and braking power assistance.

For this reason, within the framework of the safety concept, an attempt is made to assess the driving condition and the driver's reactions to the best possible extent, and thus to achieve a slower transition to the emergency program that can still be controlled by the driver.

However, this is only possible as far as the technology allows.

6.1. TRANSITION TO STAGE A - PWG EMERGENCY DRIVING WITH A PWG SENSOR

With the transition to stage A, a switch is made to a PWG emergency operation progression characteristic, which can result in a jump to a smaller PWG setpoint specification and thus a sudden change in load.

This negative jump in pedal value is therefore not forwarded directly to the torque manager, but the delta (current value - target value) is ramped down. Changes in driving are passed on immediately and unfiltered. If the driver applies the brake or the clutch during the curtailment time, the pedal value zero is output immediately.

6.2. TRANSITION TO STAGE B - PWG EMERGENCY DRIVE WITHOUT PWG SENSOR

The transition to level B is analogous to the transition to level A.

6.3. TRANSITION TO STAGE 1 - DK EMERGENCY DRIVING WITH A DK SENSOR

The emergency program stage 1 includes a torque limit and a limit of the Egas target value. Here, too, there should be no sudden torque jumps in the engine, but the engine torque should be reduced to the new target values in a gradient that the driver can control and assess.

For this purpose, similar to stage A, the maximum torque, starting from the current engine torque, is ramped down to the maximum torque of the emergency stage.

6.4. TRANSITION TO STAGE 2 - EMERGENCY DRIVING VIA IDLE CONTROL SYSTEM

The transition to the level 2 emergency program depends very much on the type of error. If, for example, there is a defect in the servomotor control, the throttle flaps are automatically closed by springs without the DME having any influence on this.

If the signals from DK sensors 1 and 2 are mutually implausible, but for which the faulty sensor cannot be determined with certainty, it may also be necessary to switch off the servomotor immediately.

In those cases in which feedback on the actual position is still available and the setpoint position can still be adjusted, the throttle flaps are not closed abruptly, but again via a ramp-shaped regulation (unless the driver requests otherwise). For this purpose, starting from the actual torque of the engine, the setpoint torque is reduced until the setpoint position specification for the throttle flaps reaches the value zero. The servomotor is then switched off and the speed and speed limitation activated.

6.5. TRANSITION TO STAGE 3 - EMERGENCY DRIVING WITH OPEN THROTTLES

The stage 3 emergency program becomes active when the actual DK position exceeds the DK target position for a defined period of time and the throttle flaps could not be closed despite energizing the servomotor in the closing direction.
Since this case can lead to unwanted vehicle acceleration, the response time of the DME to this error is relatively short. After a filter time of K_EDKSI_T_BL_AUF_R has expired, the torque interventions via ignition and injection to reduce the excess engine torque are released. If the error condition is still present after the filter time K_EDKSI_T_BL_AUF_F has expired, the Egas system switches to emergency program level 3.

In this stage, analogous to the other transitions, an attempt is nevertheless made to regulate the engine torque in a ramp-like manner and then switch off the Egas setting bridge. The actual engine torque derived from the load signal is roughly adjusted to the driver's desired torque by means of partial firing and intermediate motor retard.

Driving operation - especially when the driver specifies zero torque and when the traction is broken - depends heavily on the actual position of the throttle valves and cannot be guaranteed.

6.6. TRANSITION TO STAGE 4 - EMERGENCY DRIVE WITH SG INTERNAL ERROR

The stage 4 emergency program is always active when an internal control unit error is detected. Since the malfunction of the Egas system cannot be precisely predicted in these cases, the engine output is reduced to a safe minimum using redundant measures.

The driver of the servomotor is switched off by hardware from both processors via independent enable lines. The function computer (master processor) has two options for intervening in the torque output of the engine with the ignition and the filling control. The monitoring processor also has an effective possibility of intervention with the injection (partial firing or complete shutdown).

7. IMPLEMENTATION OF EMERGENCY PROGRAMS

7.1. LIMIT INDEXED ENGINE TORQUE

In the emergency programs of levels 1 - 4, the driver's desired torque is limited to the value KL_MD_MAX_SK (x-axis = no. of the emergency program). A transition function ensures that the limitation does not take effect suddenly, which can also cause a safety-critical driving condition, but based on the current driver's request, this is regulated to the new target value with the ramp KL_MD_GRAD_SK. The curtailment ends or is canceled when the target value is reached, the driver brakes or the DSC intervenes. If, on the other hand, the driver's request falls below the limit value, this is not interrupted, but continues to run in the background to allow the driver to lift or shift gears briefly.

7.2. TORQUE REDUCTION VIA TIMING ANGLE INTERVENTION

With the occurrence of an Egas emergency program of level 3 or 4, the ignition angle intervention path of the torque manager is released. From the relative charge measured by the HFM and the current engine speed, this calculates the momentary indicated torque that the engine delivers at the current operating point with its basic ignition angles. If the torque desired by the driver falls below the current engine torque, the excess torque is compensated for by retarding the ignition angle. At most, a retardation is possible up to the defined minimum ignition angle tz_min, which can result in a torque reduction of up to 40%.

7.3. TORQUE REDUCTION VIA INJECTION OVERRIDES

In the emergency program stages 3 and 4, a torque intervention via the injection is released parallel to the ignition angle intervention. The task of this intervention is to compensate for excess torque, which cannot be completely compensated for by the ZW retardation, by partially firing the cylinders.

For this purpose, the minimum actual torque of the engine that can be represented by means of ZW retard adjustment, calculated from a speed-load map and the minimum ZW efficiency, is set in relation to the driver's desired torque after considering all torque interventions. If the ratio falls below the value one (md_sk_soll / md_sk_ist), individual cylinders are switched off via the injection and thus the torque output of the engine is reduced in steps of 1/number of cylinders. The remaining excess torque can then be reduced again by means of the ZW intervention.

The calculation and execution of the partial firing is the responsibility of the slave processor and, apart from the calculation of the actual torque, is independent of the function computer.

7.4. VEHICLE SPEED LIMITATION

implemented but not yet documented

7.5. LIMITING VEHICLE ACCELERATION

If the Egas system is in an emergency program of levels 1-4, the maximum longitudinal acceleration of the vehicle is limited to a value defined for this level. The current longitudinal acceleration is calculated by the DSC and transmitted to the engine management system via CAN.

The acceleration limitation is set up as a PI controller. The P component is calculated from the characteristic KL_MD_SK_AX_P = f(delta_ax) and is also weighted using the gear-dependent characteristic KL_MD_SK_AX_GANG. The increment of the I controller is calculated from the characteristic KL_MD_SK_AX_IPOS = f (delta_ax). If the vehicle acceleration falls below the permissible maximum value again, the I component is reduced to zero via the KL_MD_SK_AX_INEG characteristic.

7.6. LIMITING THE ENGINE SPEED

Another safety measure consists in lowering the speed limit. For this purpose, a maximum speed is defined in the characteristic curve KL_N_MAX_SK for each emergency program level. If the engine speed exceeds this limit value, all cylinders are immediately switched off via the injection.

This safety mechanism also runs on the slave processor and is completely independent of the function computer, since the processor also has its own speed detection.

7.7. LIMITING THE EGAS ACTUATOR DYNAMICS

This measure is actually only effective in emergency program 1, since the activation of the servomotor is switched off in all other emergency programs. It is intended to limit the dynamics of the servomotor by reducing the maximum control duty cycle and thus enable easier plausibility checking of the DK potentiometer via the HFM load signal.

7.8. TURN OFF THE EGAS ACTUATOR

The Egas servomotor is switched off in parallel via three switch-off paths. • Fixed
setpoint specification = zero for Egas position controller
• Deactivating the enable line of the function calculator for the H-bridge
• Disabling the enable line of the H-Bridge monitor computer

The effectiveness of the shutdown is monitored via the HFM load signal, in which the measured air mass must not exceed a limit value that is above the value that can be reached via the idle actuator system.

8. MONITORING SENSORS / INPUTS

8.1. ANALOG SIGNALS

8.1.1. ON -BOARD POWER SUPPLY TERMINAL 87 (MAIN RELAY)

The vehicle electrical system voltage switched via terminal 87 supplies a large part of the actuators and the SG internal voltage regulator. The vehicle electrical system voltage is recorded analogously and checked for min/max values. During the start-up process, where voltage dips can occur, the lower diagnostic threshold is set to 5V, since the voltage regulator reset must be active at this value and the processors can no longer run.

When the valid range is left, an error filter is started and the supply voltage is immediately set to a substitute value (protection of the ignition output stages).

Since the sensor supply is derived from the Kl87 vehicle electrical system voltage, there is a risk that the monitoring modules of the sensors will already be active if the main relay picks up with a time delay, and will therefore detect a supply voltage error or error in the sensor, which would result in a change to the Egas emergency program . For this reason, the affected modules are only released when the supply voltages have been recognized as being present. If the supply voltage is still not present after a defined period of time, a main relay fault is detected.

8.1.2. SENSOR SUPPLY

The MSS54 has two separate 5V supply voltages Uext1 and Uext2 for the PWG and DKG potentiometers and HFMs. The sensor supply is read back and monitored in the control unit and taken into account when calculating the PWG and DK positions. If a supply voltage leaves the permissible range, an error filter is started. The Uext value is limited to the minimum or maximum value until the error filter expires. After the error filter has expired, the Uext value is set to the substitute value and all sensors connected to this supply voltage are considered faulty.

If the sensor supply Uext 1 fails, the sensors PWG1, DKG1 and the HFM also fail, so that the Egas system goes into emergency mode A - emergency driving via a pedal value sensor and into emergency mode 2 - driving via idle actuator (redundancy via HFM no longer given), changes.

If the sensor supply Uext2 fails, the sensors PWG2 and DK2 fail. The Egas system changes to emergency mode A and emergency mode 1 - emergency driving with a DK encoder (redundancy given via HFM).

8.1.3. PEDAL TRANSMITTER

For safety reasons, the detection of the accelerator pedal position is redundant. The pedal value sensor consists of two separate potentiometers with different characteristics and independent ground and voltage supplies.

The monitoring of the pedal value sensors is divided into two areas - the monitoring of each sensor channel and the comparison of the two pedal values.

Min/Max monitoring of pedal value transmitter pwg1 or pwg2

Monitoring is active as soon as the sensors are supplied. If the sensor voltage falls below a specified minimum threshold or exceeds a maximum threshold, the measured value is discarded and error filtering is started. After the error filtering has expired, the sensor is marked as faulty.

Channel comparison pwg1 to pwg2

The channel comparison has the task of monitoring the two pwg signals for their plausibility with one another. If the difference in the pedal positions exceeds a limit value, a PWG channel comparison error is recognized and error filtering is started. The permitted difference depends on the value of the smaller pwg position in order to be able to treat differences close to idle differently from differences in the full load range.

Decision matrix PWG monitoring

All diagnostic information that is relevant to the detection of the pedal value sensor is linked to one another by means of a decision matrix and a PWG operating mode and a reference sensor are determined from this. The use of a matrix has the advantage that it is complete and easy to understand and the corresponding software remains relatively simple and therefore also testable.

The following diagnostic information is taken into account as input signals in the matrix:

- Error in sensor supply pwg1
- Error in sensor supply pwg2
- Range error pwg1 confirmed
- Range error pwg2 confirmed
- Channel comparison error in the filter
- Channel comparison error confirmed

The result of the decision matrix is one of three possible PWG operating modes:

- Mode 0 : PWG module error-free
- Mode 1 : Failure of a PWG
  Change to emergency program level A
- Mode 2 : Failure of both PWG
  Switch to level B emergency program

Special case: high resistance of a potentiometer in the lower reversal point

Deposits or abrasion of the wiper track can result in high resistances at the lower reversal point, which result in the sensor signal becoming smaller. The consequence of this is that the zero point adaptation for this encoder is pulled down and the sensor signal may even wander below the minimum value. Since this effect is only limited to the bottom reversal point, but the encoder works properly in the remaining range, no emergency program should be activated in this case, but only an error memory entry for the workshop.

The PWG detection or monitoring behaves as follows in the event of high resistance at the reversal point: The zero point adaptation follows the decreasing sensor signal only up to a lower adaptation limit and then remains at this limit. At the same time, the error PWG high resistance is entered. Monitoring of the minimum value is deactivated as long as the second encoder signal is still in the idle range. When leaving the no-load range, the other encoder must also leave the high-impedance range. Otherwise, either a min/max monitoring error or a channel comparison error is detected.

Exact description of PWG acquisition and monitoring: see **module description PWG**

8.1.4. HFM-SIGNAL

The hot-film air-mass meter is monitored using min/max thresholds, within which the measured ML signal must lie.

On the other hand, a plausibility check of the HFM signal for the DK position during operation is not carried out, since the influences of air pressure, air temperature and vanos (catalyst heating, vanos error) would require the tolerance limits to be expanded too much.
If a DK encoder fails, the remaining DK potentiometer is monitored using the HFM signal. This is easier to do in this case, since the system is then in an emergency program and the engine dynamics are limited and the catalyst heating function is blocked. However, a vanos fault occurring at the same time could still lead to the tolerance band being left, but this would only result in a change to an even more severe emergency program - emergency driving via the idle actuator system.

Min-/Maxwertüberwachung:
Each calculated ML value of the HFM (in the case of the 8-cylinder: individual values of the two HFMs) is checked for the defined min/max limits. If the measured value is outside the limits, it is discarded and the ml substitute value is used instead. In addition, an error memory entry is made after the error filtering has expired.

Comparison of HFM signal with substitute value

Prerequisite: error-free HFM Error in DK system (failure of an encoder or error channel comparison)

If a DK encoder fails, the HFM signal is used to monitor the remaining potentiometer. If there is an error in the DK channel comparison, an attempt is made to localize the faulty potentiometer via the HFM signal.

For this purpose, an RF substitute value is calculated for each DK sensor, taking into account the duty cycle of the idle actuator, the intake air temperature and the ambient pressure. This substitute value is compared with the RF signal measured by the HFM. If the measured and the calculated value are within a tolerance band, the DK value is considered plausible and a flag is set in a ring buffer with 16 entries. If the number of IO flags in the ring buffer falls below one for a defined period of time specified threshold, the DK value is considered implausible and there is a change to emergency program 2 - emergency driving via the idle actuator system.

The same applies if monitoring is not possible because an HFM error has already been detected.

8.1.5. THROTTLE POTENTIOMETER

For safety reasons, the detection of the throttle position is redundant. Two separate DK encoders with mutually inverse characteristics and mutually independent ground and voltage supplies are installed.

Since the throttle valve position represents the actual value for the Egas position controller and this reacts immediately to any faulty sensor values, special attention must be paid to DK monitoring.

The monitoring of the throttle valve sensors is divided into two areas - the monitoring of each sensor channel and the comparison of the two DK values.

Min/max monitoring DK encoder dk1 or dk2

Monitoring is active as soon as the sensors are supplied. If the sensor voltage falls below a specified minimum threshold or exceeds a maximum threshold, the system immediately switches to the second measured value and error filtering is started. After the error filtering has expired, the sensor is marked as faulty and switched to the emergency program level 1.

Channel comparison dk1 to dk2

The channel comparison has the task of monitoring the two DK signals for their plausibility with one another. If the difference between the DK positions exceeds a limit value, an error in the DK channel comparison is recognized and error filtering is started. The permitted difference depends on the value of the smaller DK position in order to be able to treat differences close to idle differently from differences in the full-load range.

The case proves to be extremely problematic when both DK signals are considered plausible on their own, but the difference between them is too great. The procedure of the PWG channel comparison - use of the less critical (smaller) value - is not so simple here.
For safety reasons, the larger value for the actual position must be used for the DK channel comparison. However, if this is the erroneous value, this leads to an immediate closing of the throttle valves and thus to a spontaneous loss of engine power.

An attempt is therefore made to localize the faulty sensor signal by means of a plausibility check using the HFM signal. If it is not possible to localize the faulty encoder, the larger value will continue to be used as the actual value.

Decision matrix DK monitoring

All diagnostic information that is relevant for detecting the throttle valve sensor is linked to one another by means of a decision matrix, analogous to PWG monitoring, and a DK operating mode and a reference sensor are determined from this.

The following diagnostic information is taken into account as input signals in the matrix:

- Error in sensor supply dk1
- Error in sensor supply dk2
- Range error dk1 confirmed
- Range error dk2 confirmed
- Channel comparison error in the filter
- Channel comparison error confirmed

The result of the decision matrix is one of four possible DK operating modes:

- Mode 0 : DK module error-free
- Mode 1 : Channel comparison error - plausibility check with HFM signal not yet successful
  Switch to level 1 emergency program
- Mode 1 : confirmed failure of a DK encoder
  Switch to level 1 emergency program
- Mode 2 : Failure of both DK encoders
  Switch to stage 2 emergency program

Special case: high resistance of a potentiometer in the lower reversal point

The problem with the potentiometer high resistance in the lower reversal point is with the throttle flaps even more complicated than with the pedal value sensors. In order not to create a critical state in the event of a line tear, the SG signals must be wired internally with pull-up or pull-down resistors in such a way that a larger value is recognized as the DK value.
For high-impedance reversal points, this means that too large DK positions are also detected here. The channel comparison would detect a deviation that was too large and the comparison with the HFM signal would identify the DK sensor with the high resistance as faulty. For safety reasons, you should not try to distinguish these cases from actually wrong sensor signals, but switch off the sensor and switch to the emergency program level 1.

Exact description of DK detection and monitoring: see **module description DK**

8.1.6. COOLANT WATER TEMPERATURE (ENGINE TEMPERATURE)

The engine temperature (cooling water temperature engine outlet) is used within the torque manager to calculate the drag torque. Since this depends very much on the engine temperature, its influence on the Egas system should not be underestimated.

The motor temperature is monitored in two stages:

- Min-/Maxgrenzwerte
- Minimum engine temperature depending on starting temperature and engine running time

A slow time constant of the low-pass filter provides further security against short-term disturbances.

In the event of an error, the oil temperature is used as a substitute value above an oil temperature threshold.
Below the threshold or if the TOG fails at the same time, the intake air temperature is used as the starting value for a substitute value, which is then increased over a time ramp.

8.1.7. OIL TEMPERATURE

The influence of the oil temperature is similar to that of the engine temperature. The oil sump temperature is measured, but the engine inlet temperature is of interest for determining the friction torque.
Since the M engines have oil/water (8 cylinders) or oil/air heat exchangers (6 cylinders), the two temperatures differ greatly from one another. For this reason, models that take into account the influence of engine temperature, vehicle speed and air temperature are necessary for calculating the oil temperature.

The oil sump temperature is recorded via the thermal oil level sensor TOG. This sensor supplies a PWM signal, the frequency of which transmits the oil level and the pulse duration of which transmits the oil temperature.
Naturally, this interface is relatively insensitive to interference.

The following mechanisms are active as monitoring:

- Timeout monitoring
- Minimum or maximum pulse duration
- Min/max values of the oil temperature

In the event of an error, the engine temperature is used as a substitute value (even if the engine temperature sensor fails)

8.1.8. INTAKE AIR TEMPERATURE

The maps for determining the actual and maximum torques of the engine are based on standard conditions (air temperature 20ÿC, air pressure 960mbar). When calculating the moments, the current air temperature is taken into account in the form of a correction factor.

The intake air is measured using an NTC sensor integrated in the HFM. Monitoring takes place via a minimum/maximum value plausibility check. In the event of an error, a fixed substitute value is used and the correction factor for the torque calculation is set to 1.0.

8.1.9. AMBIENT PRESSURE

The influence of the ambient pressure on the torque calculation is analogous to that of the air temperature.

The air pressure is measured by a pressure sensor integrated in the MSS54 and /max thresholds monitored. In the event of an error, a fixed substitute value is also used and the correction factor for the torque calculation is set to the value 1.0.

8.2. DIGITAL SIGNALS

8.2.1. BRAKE LIGHT SWITCH

The brake light switch has the following influences on the Egas system:

- Switch-off condition for the cruise control
- Safety feature for PWG emergency driving
- Safety function in the Egas emergency program

In addition, the engine control for the DSC system carries out the plausibility check of the brake light switch and transmits the result to the DSC via CAN.

The information "brake actuated" is redundant in the MSS54:

- Brake light switch function calculator, read in digitally
- Brake light switch safety computer, read in digitally
- Brake test switch Function calculator, read in digitally
- DSC brake light switch, read in via CAN (can optionally be evaluated)

As soon as one of the three or four switches signals the status "brake actuated", this is considered to be actuated (OR operation - no majority decision). If the information differs for more than a defined period of time, the brake switch system is considered defective. The brake is considered permanently applied for the remainder of the driving cycle and the brake switch system error is entered.

8.2.2. BREAKDOWN SWITCH

The adhesion switch basically consists of two switches connected in series - a clutch switch and a switch in the gearbox, which detects the empty lane. The task of the switch is to detect a connected or open drive train.

The influence of the switch is manifold. The condition "no adhesion" is used as

- Switch-off condition for the cruise control
- Release condition for idle control
- Bridging the torque filter
- Blocking condition for gear recognition (no gear engaged)

The switch is monitored separately for the states closed or open. When the vehicle is stationary and the engine is running, the switch does not have to detect a positive connection. In overrun, on the other hand, the switch must detect adhesion when the engine speed remains above a threshold.

8.3. SERIAL INTERFACES

8.3.1. CAN

Monitoring CAN bus line

The CAN controller monitors the CAN bus lines directly. To do this, it leaves behind each of its sent telegrams and compares them. Furthermore, the telegrams received are monitored for their telegram format and for the checksum. If errors are detected, an internal error register is incremented. After an error threshold has been exceeded, the controller disconnects itself from the CAN and signals this to the CPU via a status bit. This status bit is read out cyclically by the CPU every 100 ms. In the event of an error, there is an error memory entry and the CAN controller

is reinitialized.

Timeout monitoring takes over the fail save for the received messages if the CAN does not work again within the timeout period.

Timeout monitoring of the received telegrams

The timeout monitoring controls the cyclic reception of the CAN telegrams. If this is not done for a telegram-specific period, an error memory entry is made and the CAN variables of this telegram are set to neutral values.

Timeout monitoring is active as soon as
- Terminal 15 on
- and vehicle electrical system voltage > K_CAN_UBMIN
- and time since last undervoltage > K_CAN_ED_TSPERR
- and time since last SG initialization > K_CAN_ED_TSPERR

The following CAN telegrams are currently being monitored

telegram	sender	Timeoutwert
ASC1	DSC	300ms
ASC2	DSC	300ms
ASC3	DSC	300ms
LWS1	Steering angle sensor	300ms
INSTR2	instrument cluster	1000ms
INSTR3	instrument cluster	1000ms

In order not to let the number of error locations get out of hand, only the absence of the CAN telegrams ASC1, LWS1 or INSTR2 lead to error memory entries, since it is assumed that if a transmitter fails, all telegrams of this transmitter will be missing.

Plausibility check of the DSC torque interventions

Since the DSC can increase and greatly reduce engine power via the torque interface, the DSC interventions must be checked for plausibility. This is done using redundantly transmitted information that must be plausible to one another. Otherwise an error filter is started, after which an error memory entry is made and any active DSC intervention is aborted.

The type of plausibility check corresponds to the scope required in CAN specifications 11H, Rev 1.4. The filter time for implausible interventions is 300ms. The alive counter for better monitoring of MSR interventions is supported by the DME. (configuration parameter K_ASC_ALIVE), however, cannot be used at the moment because the DSC3 from Bosch cannot supply it.

Cancellation of a DSC torque intervention

In the event of a CAN failure, a timeout of the ASC message or implausible interventions, any active DSC momentary intervention that may still be active is terminated after error filtering has expired. MSR interventions (torque-increasing) are aborted immediately. ASC interventions (torque-reducing), on the other hand, are ramped up to the driver's desired torque.

Protection against excessive interrupt load

The MSS54 works interrupt controlled on the receiving side. This means that each telegram received immediately results in a CPU action. This harbors the risk that the program flow in the engine control can be severely impaired by a faulty CAN participant which is constantly transmitting. In order to protect against this, a maximum interrupt load per receiving channel was defined; if this is exceeded, the receiving channel is switched off for the remainder of the engine run.

8.3.2. MFL

The MSS54 has an integrated driving speed controller FGR, which is operated by the driver via a multi-function steering wheel MFL. The MFL itself contains four buttons for operating the FGR:

- On/Off
- Sit/accelerate
- Delay
- Resumptions

Communication between the DME and MFL takes place via a unidirectional, serial, single-wire interface. To secure the communication and the transmitted data, the four button information is converted into a redundant 7-bit information and expanded by a further 24 bit, the value of which is predefined. In order to also be able to monitor the cyclic renewal of the information, another bit, the so-called toggle bit, which must change in a defined time frame, is added. In total, this results in a 32-bit data stream, which is sent cyclically every 20ms from the MFL to the DME.

The MFL monitoring within the DME is thus able to check the interface for the following errors monitor:

- Telegram timeout
- Toggle bit error (no change in the defined time frame)
- Format error of the fixed 24 bits
- Invalid combination of 7-bit button information

If the DME detects one of these error states, error filters are started. After it has expired, an error memory entry is made and any active FGR operation is aborted.

More information about the FGR module: see module description fgr.doc

9. MONITORING ACTUATORS / OUTPUTS

9.1. CONTROL UNIT (H-BRIDGE, ACTUATOR, DK-MEACHNIK)

9.1.1. ELECTRICAL DRIVER DIAGNOSTICS

The Motorola H-bridge, which controls the Egas servomotor, has a status output which is evaluated by the function computer with each controller cycle. The H bridge reports the following states via the status output:

- Undervoltage of the bridge supply
- Over temperature
- Overcurrent
- Function computer switch-off path active
- Safety computer switch-off path active
- Interruption of switch-off path function computer
- Interruption of safety computer switch-off path

In all these cases, the H-bridge switches off automatically (the outputs become high-impedance) and must be reactivated by the function or safety computer.

Since the states of undervoltage, overtemperature or overcurrent cannot be ruled out under extreme operating conditions, activation of the status output is only stored in the error memory. However, it has no effect on the operating mode of the Egas system, since the target
/Actual comparison of the Egas position covers all these cases.

9.1.2. TARGET/ACTUAL COMPARISON EGAS POSITION

The comparison of the target position of the throttle flaps with their actual position is one of the most important monitoring functions in the Egas safety concept. It shows the following errors:

- Processor modules
  - CTM module (processor): generates control duty cycle for servomotor
  - Processor port C: Direction of rotation of the servomotor
  - Processor port C: Enabling servomotor function computer
  - Processor port C: Enabling of the safety computer servomotor
- H-bridge actuator
  - H-bridge defect
  - Over-temperature shutdown
  - H-bridge current limit
  - H-bridge overcurrent shutdown
- Wiring servomotor
  - Line break
  - Short-circuit to ground, Ub, or between the lines
- Actuator
  - Winding defect
  - Mechanical damage
  - Transmission damage
- DK-Kinematics

- Mechanical damage

- Throttle valves
  - party-clamping flap

- Throttle valve adaptation
  - Shift of the zero point
  - Displacement of the anchor point

Case 1: The throttle flaps should be opened above a threshold, but the flaps remain closed.

Reasons: Processor module defective
H-bridge defective or switched off for a short time
Safety shutdown activated
actuator wiring
Servo motor defective
DK-Kinematics defective

Error detection:
Egas target position > K_EDKSI_POS_ZU + K_EDKSI_HYS_ZU
and Egas actual position < K_EDKSI_POS_ZU
for Time > K_EDKSI_T_ZU

Reaction: Change to Egas emergency program stage 2 - driving via idle controller

Evaluation: The throttle flaps remain closed or are closed independently by the spring packs without the control unit being able to influence this.
The reduction in torque when the flaps are closed cannot be influenced either (critical condition for case 1). If the flaps are closed, it is possible to continue driving in the emergency program without any problems if it is ensured that the flaps can no longer open.

Case 2: The throttle valves should be closed, but remain open a little.

Reasons: Throttle valve is stuck or extremely stiff
slight twisting of the control potentiometer of the throttle valve system
incorrect zero point adaptation

Error detection:
Egas target position =
0 and K_EDKSI_POS_ZU < Egas actual position < K_EDKSI_HYS_BL_AUF
for Time > K_EDKSI_T_SPALT

Reaction: no Egas emergency program - retention of the current operating level
Error memory entry

Assessment: The fact that the throttle flaps remain open a little despite being pressed by the servomotor indicates a major problem in the DK system, which justifies a fault memory entry. However, the K_EDKSI_HYS_BL_AUF limit is dimensioned in such a way that it would result in an increased idle speed when idling, but is not regarded as safety-critical for driving operation and therefore no change to an Egas emergency program is necessary.

Note: By observing the idle speed, a distinction could be made between a mechanical problem and an adaptation problem:
LL speed can be adjusted: adaptation problem
LL speed too high : Flap problem

Case 3: The throttle flaps should be opened, the flaps react but do not reach the setpoint.

Reasons:
H-bridge temporarily switched off
stiff DK system
Throttle stuck below target position
undervoltage

Error detection:
Egas target position - Egas actual position > K_EDKSI_HYS_U_SOLL
and K_EDKSI_POS_ZU < Egas actual position <= K_EDKSI_POS_N_GANZ
and ub > K_ED_UBMIN
for Time> K_EDKSI_T_U_SOLL

Reaction: Change to Egas emergency program stage 2 - driving via idle controller

Evaluation: Since the reliability of the Egas system can no longer be guaranteed, the flaps are closed in a targeted manner and the control is then deactivated. If the flaps are stuck, a change to case 2 or 5 is possible as soon as the setpoint is below the actual value, or case 4 depending on the actual position (does not open completely).

Case 4: At full load, the throttle flaps do not fully open

Reasons:
Flaps on VL stop - wrong adaptation
undervoltage

Error detection:
Egas target position - Egas actual position > K_EDKSI_HYS_N_GANZ
and Egas actual position > K_EDKSI_POS_N_GANZ
and ub > K_ED_UBMIN
for Time> K_EGAS_T_N_GANZ

Reaction:
no Egas emergency program
Switching the slope of the DK potentiometer characteristic to a defined maximum slope (servomotor protection)
Start of a new VL adaptation in the run-on
Error memory entry

Evaluation: This case only results in a loss of performance in the full load range and is therefore not critical to safety. However, measures must be taken to protect the servomotor.

Case 5: The throttle valves are stuck in the open state

Reasons:
Defective processor module - 100% activation, wrong direction of rotation
H-bridge alloyed
Short circuit in actuator wiring
stiff DK system
Throttle valve is stuck above target position

Error detection:

Egas actual position - Egas target position > K_EDKSI_HYS_BL_AUF
for time > K_EDKSI_T_BL_AUF_R (detection and reaction time)
or time > K_EDKSI_T_BL_AUF_F (error filter time)

Reaction: After the detection time has elapsed, due to the possible effects of the error, torque-limiting measures are taken immediately via ignition angle intervention and injection suppression.

After the error filter time has expired, there is a switch to the Egas emergency program Stage 3 - open throttle driving

Evaluation: In this case, the engine generates more power than the driver desires and unwanted vehicle acceleration can occur. This requires a quick response to this condition. However, the control unit has the option of throttling the engine power to a range specified by the driver via ignition angle intervention and cylinder suppression.

9.2. IDLE ACTUATOR

With the idle actuator system, the engines from M GmbH also have a second air supply system that is independent of the Egas system. The maximum air flow through the idle actuator is approx. 100 kg/h compared to the 1200 kg/h through the throttle valves. The maximum speed that can be achieved with the engine at operating temperature and the drive train open is approx. 3000 rpm, the maximum speed in 6th gear, on a level route and a long start-up distance is approx. 80 km/h.

The driving performance that can be achieved in this way is classified as manageable by the driver, so that emergency driving via the idle actuator system is still permitted in the event of all errors - including internal SG errors.

The idling actuator system itself consists of a two-winding ZWD rotary actuator with an opener and a closer winding, which is connected to terminal 87 via a common supply line. If both windings are currentless, an emergency air cross-section is set via an internal spring, which roughly corresponds to a control duty cycle of 30%.

The DME controls it via two PWM signals, whereby the make winding is operated with the inverse signal of the break winding. The drivers used for the control can be diagnosed and monitor the control line with regard to

- Line break
- Short circuit to ground
- Short circuit to Ub

After detection of an electrical fault, there is an immediate reaction in the control of the ZWD. An error is stored in the error memory of the DME after an error filter has expired.

The reactions to all possible error combinations should, as far as possible, dampen the effects on engine operation and are stored in a 4x4 matrix. In the event of a short-circuit in a control line to ground, the remaining winding is also fully energized, resulting in an effective control ratio of approx. 50%. If a line fails (interruption or short-circuit to Ub), the remaining winding is operated with a minimum pulse duty factor and an opening cross-section in the area of the emergency air cross-section is established.

The allocation of the target filling to idle actuator and throttle valve, as well as the calculation of the ml substitute values, takes into account the emergency measures in the idle actuator control.

10. MONITORING CONTROL UNITS HARDWARE

10.1. PRE DRIVE CHECK CONTROL UNIT

10.1.1. MEMORY TESTS

In the initialization phase of the control unit, the two internal RAM memories of each processor are subjected to a complete read/write test. If a RAM error is detected, an internal SG error is recognized immediately and the system starts in the level 4 emergency program.

A checksum check of the program and data memory is not usually carried out during the initialization phase of the control unit, since these tests would delay the engine start unacceptably.
However, if a corresponding error was noted in the previous operating cycle of the control unit, these tests are also carried out completely again in the initialization phase. If the error is confirmed in this way, there is also a change to emergency program 4.

For more information on the memory tests, see the module description: sk_check.doc

10.1.2. PROCESSOR SYNCHRONIZATION

The MSS54 is a two-processor system, with both processors taking on around 50 percent of the engine control functionality. Communication between the two processors takes place via a Dual Ported RAM (DPR). Furthermore, the two processors are coupled via a high-priority interrupt line, which enables each processor to trigger a non-maskable interrupt for the partner.

Another level of security is that the reset inputs of the partner's processors and ports are routed so that one processor can reset the other if necessary.

Processor synchronization on SG initialization

The problem with the initialization of the control unit is that the processors communicate via a dual-ported RAM. However, since variables from the other processor are already being accessed during the initialization of the individual software modules, it must be ensured that the corresponding variables in the dual-ported RAM are already pre-initialized with sensible values. The DPR, on the other hand, cannot be initialized from one side, since this means that if one processor resets unexpectedly, it would also reinitialize the variables of the other processor.

For this reason, a synchronization level was introduced in the initialization phase of the individual processors, which is intended to ensure that the processors only start initializing the function modules when both sides have initialized their DPR variables.

The synchronization is implemented via the Inter-Processor Communication (IPK) module of the OSKAR operating system. The IPK is a communication channel secured by handshake mechanisms, checksum and timeout monitoring, which can transmit commands and data to the partner processor and which has an execution status reported back to it.

During initialization, each processor sends a synchronization request to the partner via the IPK. If this has already initialized its DPR sizes at this point in time, an OK status is reported back. If initialization has not yet taken place, there is no response. The sync request sender is now waiting for the OK status. If this is not recognized within the IPK's timeout period of currently 32ms, it repeats the synchronization attempt up to four times. If these also remain unanswered, the processor initializes to the partner's DPR sizes with neutral ones values and continues the program flow. Motor operation remains blocked until communication between the two processors has been established.

If a processor is reset during operation, it must also be resynchronized with the processor that continues to run normally during initialization.

10.1.3. PRE DRIVE CHECK EGAS ACTUATOR

The Egas control unit pre-drive check has the following tasks.

- Phase 1 : Zero point adaptation of the throttle valve potentiometer
- Phase 2 : Checking the freedom of movement of the flaps and the Egas control circuit
- Phase 3 : Checking the Egas safety shutdown of the monitoring computer and checking the flap return springs

The pre-drive check is carried out after each power-on of the control unit as soon as the supply voltage for the drivers and sensors is available.

Phase 1 is always carried out. In phases 2 and 3, the pre-drive check is aborted as soon as terminal 50 becomes active, the engine speed is not equal to zero or the vehicle is moving.

Phase 1: Zero point adaptation of the throttle valve potentiometer

After each power-on of the control unit, an adaptation run must be carried out to determine the zero point position of the throttle valve potentiometer. This is necessary because the DK encoder signal represents the actual value for the Egas control circuit and if the zero point adaptation is incorrect, the throttle flaps could no longer be closed correctly, or incorrect diagnoses could be made by the DK monitor.

The adaptation takes place when the servomotor closes the throttle flaps with a defined force.
The potentiometer voltage is then recorded several times, and if all measured values are plausible, the new zero point position for each throttle valve potentiometer is determined from this by averaging.

Details on the adaptation process can be found in the throttle valve module description.

Phase 2: Checking the free movement of the flaps and the Egas control circuit

In phase 2, the freedom of movement of the throttle valves and the adjustment behavior of the Egas control circuit are checked.

For this purpose, the setpoint egas_soll is set to the value K_PDR_EDK_SOLL. At the same time, the target/actual comparison of the Egas system and the diagnosis of the throttle valve potentiometers, including the channel comparison, are activated. After the waiting time K_PWD_T_PHASE2 has expired, the information from the corresponding monitoring modules is evaluated. If the Egas system works correctly, the target position should be adjusted and all diagnoses should report an OK status.

The following diagnoses are evaluated in detail for the Pre Drive Check - Phase 2:

- Target/actual comparison of the throttle valve position
- Sensor diagnosis DK1 encoder

- Monitoring sensor supply DK1
- Sensor diagnosis DK2 encoder
- Monitoring sensor supply DK2
- Channel comparison DK1/DK2 value

Table: Evaluation of diagnostic information Pre Drive Check

Should, is comparison	channel comparison	Diagnose DK1	Diagnose DK2	evaluation	branch into emergency program
0	0	0	0	System in Ordung	--
0	x	0	1	Failure DK2	step 1
0	x	1	0	Failure DK1	step 1
0	1	0	0	DK1 to DK2 implausible	Level 2
0	x	1	1	combination impossible	Level 4
1	0	0	0	Actual position is not reached Actual position too small Actual position too large	 Level 2 level 3
1	1	0	0	Actual position is not reached DK1 to DK1 implausible	Stufe2
1	x	0	1	Actual position is not reached Failure DK1	Stufe2
1	x	1	0	Actual position is not reached Failure DK2	Stufe2
1	x	1	1	Failure of both encoders	Level 2

0 := OK
1 := faulty
x := not relevant

Phase 2 is immediately aborted in the event of a start attempt (terminal 50 active or engine speed not equal to zero or vehicle speed not equal to zero).

Open points: Waiting time may depend on the engine temperature

Phase 3: Checking the safety shutdown and the closing springs

The task of phase 3 is to check the switch-off path of the safety computer for the H-bridge and the closing springs of the throttle valves.

For this purpose, the setpoint value K_PDR_EDK_SOLL is also specified by the function computer for the throttle valves. At the same time, the safety computer is requested to activate its switch-off path for the H-bridge. In the error-free state, the throttle flaps should now be pulled shut by the spring assemblies. If the actual position does not fall below a specified threshold within the time K_PDR_T_PHASE3, the setpoint is set to zero and the H-bridge remains switched off.

If the throttle flaps can now be closed, the switch-off path is not working. An SG-internal error is entered and a branch is made to the level 2 Egas emergency program. If the flaps remain open, the safety shutdown is then deactivated again. If the flaps can now be closed, the closing springs are defective. The corresponding error is entered and a branch is also made to the stage 2 emergency program. If the flaps remain open, the stage 3 emergency program is activated.

Phase 3 is immediately aborted in the event of a start attempt (terminal 50 active or engine speed not equal to zero or vehicle speed not equal to zero).

10.2. MONITORING THE CONTROL UNIT DURING OPERATION

10.2.1. MEMORY TESTS

During operation, the program, data and variable memories of the DME are subjected to a permanent, cyclical test. The RAM memory is checked using a read/write test, while the ROM memory (program and data) is monitored using CRC16 checksums.

If an error is detected and confirmed here, the system switches to emergency program 4 - SG internal error.

The DPR has a special position in the memory test. Since this memory is accessed asynchronously from two sides, a read/write test is out of the question here. A detection of faulty memory cells is therefore not possible. This counteracts the safety concept by keeping the DPR free of safety-critical variables. This means that all variables relevant for the filling of the engine and thus for the torque output are in an internal memory of the processor, which is subject to the RAM test, and the DPR only has copies of these values, with the copies only being used for non-critical program parts .

In cases where a security-critical exchange of values via the DPR is also necessary, this does not take place directly by storing these values in the DPR, but via the checksum-protected transport mechanism of the inter-processor communication.

10.2.2. MONITORING HW INITIALIZATION

implemented but not yet documented

10.2.3. PROCESSOR COMMUNICATION

The processor communication and its operational readiness are monitored via two
The control function.

A very simple but nevertheless very effective monitoring function consists in checking the two system timers for synchronization. To do this, each processor stores a copy of its system timer in the DPR.
If a processor does not detect any change in the timer of the partner processor for a period of K_PCNTRL_TIMEOUT, then a problem in its program processing is inferred, and the system is reset and reinitialized by means of a reset.

A second, somewhat complex control mechanism monitors the exchange of security-critical variables via the IPK. As already mentioned, this exchange mechanism works with secured telegrams, the security mechanisms of which include the following:

- Checking the telegram identifier
- Verification of the telegram checksum
- Confirmation of correct reception of the telegram
- Return value of the evaluation function of the telegram to the sender
- Timeout monitoring on the sender side with regard to receipt

If there is no problem-free communication between the two processors for a period of K_SK_IPK_TIMEOUT, the system is also reinitialized by means of a reset.

10.2.4. PROGRAMMABLAUFKONTROLLE

Each processor of the MSS54 has a processor-internal hardware watchdog. This must be served at least once from the background task (slowest task) and the 10 ms task (most important task for the Egas system) within the watchdog time of one second.

In order to be able to additionally ensure the execution of all program parts relevant for the Egas system, a program execution control was implemented parallel to the hardware watchdog. This is called cyclically by the watchdog-monitored 10ms task and checks whether all the functions relevant to the Egas system have been executed at least once within an applicable period of time.

This is implemented using a flag variable in which one bit is reserved for each function and which is set when the function is executed. If the program flow control recognizes that one of these bits is not set, an error memory entry is made and the processor is reset. If this condition occurs several times during engine operation, the Egas system goes into the stage 2 emergency program -
Emergency driving via idle actuator.

The following modules are currently monitored:
master processor :
- Detection of pedal value sensor
- Monitoring pedal value sensor
- Detection of throttle valve potentiometer
- Throttle valve potentiometer monitoring

- Target/actual comparison Egas position
- Main function security concept

10.2.5. RESET MONITORING

A number of monitoring mechanisms are implemented in the MSS54, which trigger a reset and thus cause the system to restart. Examples of such monitoring functions are:

- internal watchdog
- Occurrence of an uninitialized interrupt
- Errors in the program processing (zero part, bus error, illegal opcode, ...)
- Processor communication timeout
- Errors in the test calculations
- Timeout in the program flow control

In normal operation, however, the system should run reset-free. However, if the system reset frequency exceeds a defined limit value during an operating phase, this indicates a serious problem within the DME. However, since the reason for the problem and its effects cannot be foreseen, there is a change to the Egas emergency program 4 - SG internal error for security reasons.

The reset line of each processor is routed to an interrupt input of the partner for reset monitoring. This enables the partner to immediately recognize any reset, document it and take appropriate protective measures until the system is ready for operation again.

10.3. MONITORING THE CONTROL **UNIT IN THE FOLLOW-UP PHASE**

10.3.1. MEMORY TESTS

A complete checksum test of the program and data memory is carried out in each follow-up phase of the control unit. If an error is found, this is noted and the complete test is repeated in the next initialization phase of the control unit.

11. LOGICAL MONITORING FUNCTIONAL CALCULATOR

11.1. PROTECTION TORQUE CALCULATION

The main path of the torque calculation and all offset torques of other modules affecting it are checked for plausibility within the torque manager. If an implausible value is detected, this value is immediately converted into a neutral value and an error filter is started. After the error filtering has expired, the Egas monitoring function is notified, which then switches the Egas system to emergency mode 2 - emergency driving via the idle actuator system.

When the efficiency is corrected (ignition angle, lambda) within the torque manager, the efficiency is only limited downwards, but no error entry or change to an emergency program, since it cannot be ruled out that the limit value can be undershot in normal operation.

Security queries (error conditions):

- Indicated engine drag torque "md_ind_schlepp" < 0
- Minimum indicated engine torque "md_ind_min" > maximum indicated engine torque „md_ind_max"
- Motor torque loss > K_MD_SK_MAX_MDMIN above the speed threshold K_MD_SK_N_MDMIN
- Indexed desired torque "md_ind_ Wunsch" > maximum torque "K_MD_SK_MAX"
- Output MD dynamic filter > maximum torque "K_MD_SK_MAX"
- Resulting desired moment "md_ind_ Wunsch_red_korr" > K_MD_SK_MAX
- Desired torque for ignition angle path "md_ind_ Wunsch_tz_red" > K_MD_SK_MAX
- Target filling "md_rf_soll" > maximum filling "K_MD_RFMAX"
- Lambda lean factor > 2 (overflow)

Monitoring torque interventions

- Intervention I component of idle speed control "md_llri" > maximum intervention "K_MD_SK_LLR_MAX"
- Intervention PD component of idle control "md_llrp" > maximum intervention "K_MD_SK_LLR_MAX"

11.2. MONITORING SET TORQUE TO ACTUAL TORQUE

A plausibility check of the actual engine torque for the driver's desired torque over the entire operating range is very difficult, since in this case a large number of input parameters, all transient states and all torque interventions from other modules would also have to be taken into account.
This would require almost the entire calculation path to be stored again redundantly, which is not possible due to a lack of resources, or the corresponding tolerance limits would have to be greatly expanded.

Two torque monitoring functions have therefore been implemented in the MSS54. A function that compares the actual torque with the desired torque, taking into account all torque interventions, and has wider tolerance limits. And via torque monitoring, which is limited to a zero torque specification by the driver (PWG = zero), but is activated there accordingly. The advantage of this is that the torque calculation can be estimated much better at this operating point, and the tolerance limits can therefore be tightened. Furthermore, it can be assumed that if the engine delivers an undesirably high torque, the driver will automatically take his foot off the accelerator and the activation conditions for this test are therefore met.

11.2.1. MONITORING SET/ACTUAL TORQUE ACROSS THE ENTIRE OPERATING RANGE

Definition of the actual torque md_sk_vergl_ist =

md_ind_ne — actually generated indexed actual torque of the engine, determined from the map of speed and load (n, rf) and ZW efficiency under consideration of all interventions

Definition of the target torque md_sk_vergl_soll =

md_fw_filter — filtered driver's desired torque from PWG position or cruise control

+ md_ind_min_ges + — Motor torque losses including all consumers
md_ar + md_llri + — Intervention torque of the anti-judder control
md_llrp — Intervention torque of the I controller of the idling control
Intervention torque of the P controller of the idle speed control

In the case of a torque-increasing MSR intervention, the maximum of the required torque and md_sk_vergl_soll aks the desired torque is used.

If the actual torque of the engine exceeds the target torque for the period K_MD_SK_TIMER_MD by the amount K_MD_SK_OFFSET + (1 - K_MD_SK_GEWICHTUNG) * md_sk_vergl_ist, an error is concluded and a change to the Egas emergency program 2 driving via the idle actuator system takes place.

11.2.2. MONITORING SET/ACTUAL TORQUE WITH PWG DEFAULT = 0

Activation condition for monitoring

Operating status Engine is running
no FGR operation
no MSR intervention
Dashpot function of the dynamic filter limited
Pedalwertvorgabe <= K_MD_SK_PWGMAX

If, in this case, the calculated torque desired by the driver exceeds the value K_MD_SK_FWMAX or the calculated target DK position exceeds the value K_MD_SK_WDK_MDMIN for the period K_MD_SK_TIMER, an error in the torque calculation is concluded and the Egas system also switches to the stage 2 emergency program.

12. LOGICAL MONITORING SECURITY COMPUTER

When defining the security concept, the following philosophy was advocated:

All errors in the sensors, actuators or torque calculation should be recognized by the function computer itself and a non-critical state should be achieved through appropriate measures.

The task of the safety computer is to monitor the function computer for its operability if its own mechanisms do not recognize this. In addition to the communication tests already explained and the reset monitoring, these monitoring functions of the safety computer also include the monitoring of the analog/digital converter and the computer core of both

processors.

12.1. MONITORING ADC FUNCTIONAL COMPUTER

This test is intended to monitor the functionality of the analog/digital converter ADC of each processor. For this purpose, two analog signals - PWG1 and DKG1 - are routed in parallel to the ADC of the two processors, and are read in cyclically by them. The converters of the two processors should therefore deliver the same result.

If the difference between the two results exceeds a limit value for a defined period of time, this is interpreted as a problem with one of the AD converters, an SG-internal error is filed and the corresponding emergency program is switched to.

In order to take into account the runtime differences between the two processors, the test is hidden if both AD converters detect that the analog value is too dynamic.

Note: currently only one analog signal - PWG1 - is used for monitoring

12.2. MONITORING COMPUTER CORE

Both computer cores are monitored by means of test calculations, which are executed in parallel in both processors and the results of which are checked by the safety computer for consistency.

For this purpose, 14 test tasks are currently defined with the following main points:

Test results 1: 2: 3: Map interpolation of the unsigned short type
Characteristic interpolation of type singed short
Map interpolation of the type signed char
4: Characteristic interpolation of the unsigned char type
error filtering
error entry
error healing
error discharge
CPU test: focus on arithmetic and logical operations
5: CPU test: focus on bit operations and jump instructions
6: CPU test: focus on address arithmetic
7: CPU test: unused
8: low pass filter
9: 10: 11: 12: 13: 14: unused

These 14 test tasks are calculated cyclically with 11 different sets of parameters, resulting in a total of 154 different tasks.

The process of the computer core test runs in principle according to the following scheme:

- the safety computer selects a test task and a parameter set
- the safety computer calculates the test task and saves the result
- The selected task is transferred to the function computer for processing via IPK in the form of a task number and a parameter set number
- the function calculator calculates the result of the test task and sends it with the Acknowledgment of receipt from the IPK back to the security computer
- the safety calculator compares the two results

A test calculation is considered incorrect if the results do not match. In this case, the test task is repeated up to K_SK_TR_MAX times with the same parameter set. If the results still differ, an error is filed and the system is re-initialized by means of a reset.

As only the function computer is controlled by the safety computer through this mechanism, another feature was implemented in this test, through which the function computer has the option of also ensuring the correct processing of the monitoring function on the safety computer.
For this purpose, the function computer deliberately returns an incorrect calculation result to the safety computer with every K_SK_TR_MANIPULATION-th test calculation. This must recognize the incorrect result and repeat the test task with the same set of parameters. If this is not the case, an error in the program processing is also concluded and the system is reset using a reset.

Errors in the transmission of the test invoice are treated as communication errors.

12.3. FGR SHUTDOWN MONITORING

In FGR operation, no plausibility check between the driver's request (accelerator pedal position) and the actual torque of the engine is possible, since the target torque specification is determined by the vehicle speed controller and can be between 0 and 100% of the possible engine output. In order not to have to completely exclude this operating state from torque monitoring, a monitoring function is implemented on the safety computer, which controls the switch-off of the FGR when the brake is actuated.

The basis for the monitoring is the assumption that the driver will react to an unintentional acceleration of the vehicle in FGR mode by actuating the brake. In this case, the FGR operation must be aborted immediately and the implemented comparisons of the driver's request to the actual torque become active again.

This switch-off condition via the brake actuation is monitored by the safety computer. If the latter recognizes that the FGR operation is not aborted despite the brake being actuated, the latter concludes that the FGR function on the function computer is no longer running properly. It thus stores an error in the error memory and switches to emergency program 4 - SG internal error.

PROJECT: MSS54

CHAPTER: 4.02

MODULE: INJECTION

FUNCTION: INJECTION TIME CALCULATION

SUB-FUNCTION: SEQUENTIAL INJECTION MASS AND INJECTION TIME

AUTHORISATION

AUTHOR (ZS-M-57)________________________ DATE __________

APPROVED (ZS-M-57)__________________ DATE __________

APPROVED (EA-E-2) ____________________ DATE __________

	Department	date	Name	filename
Author	ZS-M-57	02.08.04	Erdl	4.01

Table of Contents

CHANGE DOCUMENTATION FROM R360

Version	date	comment
r360	1.6.2001	specific v. FH Mayer and documentation from MSS54 project merged
R380	29.10.2001 rm :	Modification of the nomenclature of the injection correction factors from FH Mayer
R380	13.11.2001 ke:	Anzeigevariable ti_eff_out

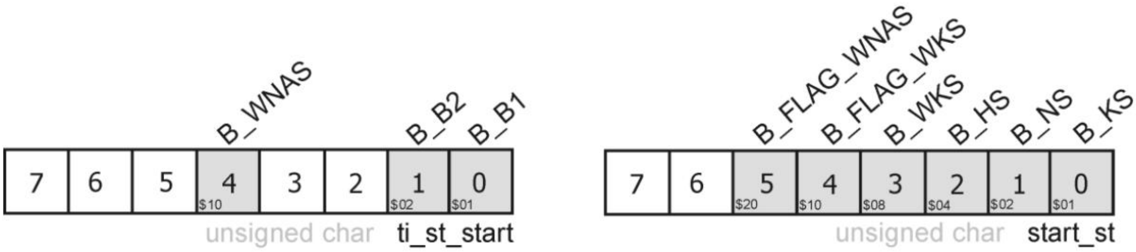
1 FUNCTIONAL DESCRIPTION

1.1 PHYSICAL BACKGROUND

In the injection module, the associated fuel mass is determined based on a cycle-consistently specified air mass for the working cycle. The basic injection mass is calculated for a target total fuel mass, taking into account correction parameters. This variable is then used for fuel balancing in the injection operating mode transitions module. After the adaptation values and component corrections have been taken into account, the injection time is then calculated.

1.2 CALCULATION OF THE CORRECTION FACTORS

The operating status is documented via status bytes:



[File : st_bytes.gif]

1.2.1 BASE ADJUSTMENT FACTOR CALCULATION

The constant K_TI_MK_GA can be specified via the application system as a multiplicative intervention on the fuel mass. It should be noted that this constant is neutral for normal operation.

(1) ti_mk_f_ga = K_TI_MK_GA

1.2.2 CALCULATION OF THE START FACTOR

The start factor is only required in the START operating state. The calculation takes place from engine standstill (B_MS), so that a valid value is already present at the transition to start B_START. This factor is determined as long as you are in START mode.

- There are conditions that must be taken into account when calculating the ti_mk_f_start factor:

  Heißstart B_HS (tmot > K_TI_MK_TMOT_HS),
  Normalstart B_NS (K_TI_MK_TMOT_KS<=tmot<=K_TI_MK_TMOT_HS),
  Cold start B_KS and
  Repeat cold start B_WKS.

These conditions are checked and set in the ti_set_startbereich() function.

- The determination of the switching conditions for the starting range of Area1 in the area2 in the start are defined as follows:

  B_B1 to B_B2 IF

  n > KL_TI_MK_TMOT_B2
  OR
  ti_anz_seg_zaehler> K_TI_MK_KW.

These conditions are checked and set in the TI module when Start is entered.

- A repeat cold start is defined as follows:

  B_WKS = 1 if tmot
  < K_TI_MK_TMOT_KS AND
  B_FLAG_WKS was set in the previous engine run
  AND service life t_motor_stands
  < KL_TI_MK_WKS_MS_TMOT
  OTHERWISE
  B_WKS = 0
  B_KS = 1.

The repeat cold start flag B_FLAG_WKS (BIT4 in start_st) is set when

  the engine is switched off (B_KLA)
  AND the engine was switched off in the starting area B_B2
  OR the total engine running time was less
  than KL_TI_MK_WKS_ML_TMOT,
  ELSE
  B_FLAG_WKS is deleted.

It is then saved in NVRAM.

1.2.2.1 Hot start and range2 (B_HS and B_B2)

(2) ti_mk_f_start = ti_mk_f_n_ks (KL_TI_MK_N_KS) *
ti_mk_f_tan_hs (KL_TI_MK_TAN_HS)

1.2.2.2 Hot start and !Range2 (B_HS and !B_B2)

(3) ti_mk_f_start = ti_mk_f_tan_hs (KL_TI_MK_TAN_HS)

1.2.2.3 !Hot start and area 2 and !Repeat cold start (!B_HS and B_B2 and !B_WKS)

(4) ti_mk_f_start = ti_mk_f_n_ks (KL_TI_MK_N_KS)
*ti_mk_f_tmot_ks(KL_TI_MK_TMOT_KS)
* ti_mk_f_kw_zaehler (KL_TI_MK_KW)

1.2.2.4 !Hot start and range2 and repeat cold start (!B_HS and B_B2 and B_WKS)

(5) ti_mk_f_start = ti_mk_f_n_ks (KL_TI_MK_N_KS)
* ti_mk_f_tmot_ks(KL_TI_MK_TMOT_KS)
* ti_mk_f_kw_zaehler (KL_TI_MK_KW)
* K_TI_MK_WKS_B2

1.2.2.5 !Hot start and !Range2 and !Repeat cold start (!B_HS and !B_B2 and !B_WKS)

(6) ti_mk_f_start = ti_mk_f_tmot_ks(KL_TI_MK_TMOT_KS)

1.2.2.6 !Hot start and !Range2 and repeat cold start (!B_HS and !B_B2 and B_WKS)

(7) ti_mk_f_start = ti_mk_f_tmot_ks(KL_TI_MK_TMOT_KS)
* K_TI_MK_WKS_B1

1.2.3 CALCULATION OF THE FACTOR IN STATIONARY OPERATION

The factor ti_mk_f_stat is multiplied by the fuel mass as a stationary lambda correction value.

1.2.3.1 Vollast

(8) ti_mk_f_stat = KF_TI_MK_N_WI_VL

1.2.3.2 All other operating states

(9) ti_mk_f_stat = KF_TI_MK_N_WI

1.2.4 CALCULATION OF THE KAT PROTECTION FACTOR

When activated, the cat protection factor is always **>= 1.0** and depends on the ignition angle retard.
Cat protection is implemented via a pre-control and an I controller. As soon as the cat protection factor is > 1.0, ie the cat is cooled, the lambda control is deactivated.

1.2.4.1 Pre-control

The entry condition for calculating a precontrol value not equal to one is met when the pullback ignition angles from the knock control and the knock adaptation assume negative values. Only then is the pre-control determined bank-selectively:

(10) dtz_sum[j] = kr_dtz_sum[j] + ka_dtz_sum[j]

with j = 1, 2 (Bank-j)

Here dtz_sum[j] is the sum of all retraction angles related to a bank and always has a numerical value less than zero.

The ignition angle offset ti_mk_tz_offset_kats is applied as a threshold value for calculating the pre-control value.

ti_mk_tz_offset_kats =
IF VL
KL_TI_MK_KATS_VL_N
OTHERWISE
KF_TI_MK_KATS_N_WI.

This results in:

(11) temp[j] = (-1) * (dtz_sum[j] + ti_tz_offset_kats)

(12) ti_mk_f_kats_steuer[j] = 1 + (temp[j] * K_TI_MK_KATS)

with j = 1, 2 (Bank-j)

If the difference between the sum of the retraction angle and the offset value is positive, Eq. (11), the pre-tax factor ti_mk_f_kats_steuer[j] = 1.0 is set, otherwise the multiplication takes place by minus one and the inclusion in Eq. (12).

1.2.4.2 I-Rules

In order to activate the I controller, a wi threshold must be exceeded. This is to avoid an unnecessarily long enrichment.

The release condition is met when:

wi > KL_TI_MK_KATS_WI_SCHW_N

If this release condition is not met, ti_mk_f_kats_regler = 0 is set.

The I controller is implemented using a state machine whose state variable is the exhaust gas temperature TABG. The exhaust gas temperature must exceed a threshold for the controller to be activated:

TABG> = K_TI_MK_KATS_TABG_EIN

The status KATS_AKTIV is set as a result.

Status CAT_ACTIVE:

As long as the exhaust gas temperature exceeds the switch-on threshold (K_TI_MK_KATS_TABG_EIN), the controller value is calculated as follows:

(13) ti_mk_f_kats_regler (k) = ti_mk_f_kats_regler (k-1)
+ KL_TI_MK_KATS_DELTA_ML

The next state is reached when the exhaust gas temperature exceeds a next higher threshold.

TABG> = K_TI_MK_KATS_TABG_SCHNELL

As a result, the state KATS_SCHNELL is set.

The curtailment state is reached when the curtailment threshold is not reached.

TABG <= K_TI_MK_KATS_TABG_AUS

The status KATS_ABREGELN is set as a result.

However, if the exhaust gas temperature is between the up-regulation threshold and the down-regulation threshold, the controller is stopped to prevent an overflow (integrator stop).

Status CAT_FAST:

In this state, an override is generated with the help of a factor.

(14) ti_mk_f_kats_regler (k) = ti_mk_f_kats_regler (k-1)

+ (KL_TI_MK_KATS_DELTA_ML
* K_TI_MK_KATS_FAK_SCHNELL)

In the slow Aufregelbereich you get back when the exhaust gas temperature threshold

TABG <K_TI_MK_KATS_TABG_SCHNELL

falls below This again corresponds to the state KATS_AKTIV.

Status KATS_CUT DOWN:

In the following state, the controller is regulated back to zero because the exhaust gas temperature has fallen below the switch-off threshold that can be applied.

(15) ti_mk_f_kats_regler (k) = ti_mk_f_kats_regler (k-1)
- KL_TI_MK_KATS_DELTA_ML

However, if the exhaust-gas temperature threshold rises above the up-regulation threshold during this process, the system changes back to the KATS_ACTIVE state.

1.2.4.3 Total Enrichment Factor

The following factor is included in the injection mass equation (Section 4.2, Eq. (7)),

(16) ti_mk_f_kats [j] = ti_mk_f_kats_steuer [j] + ti_mk_f_kats_regler

and with j = 1, 2 the bank-selective influence is taken into account. The total enrichment factor is limited to K_TI_MK_F_KATS_MAX before the calculation.

1.2.5 POST-START FACTOR CALCULATION

The calculation is carried out in the 10 msec task. The post-start factor is regulated using an exponential function. The starting value for the exponential function is determined during the transition from the START operating state to MOTOR RUNNING.

If the restart factor is less than the threshold K_TI_MK_SCH_NAS, the Time constant ti_mk_tau_nas calculated as follows:

(17) ti_mk_tau_nas = KF_TI_MK_TAN_TMOT_NAS
* K_TI_MK_TAU_NAS

If the post-start factor is greater than or equal to the threshold K_TI_MK_SCH_NAS, the time constant ti_mk_tau_nas is calculated as follows:

(18) ti_mk_tau_nas = KF_TI_MK_TAN_TMOT_NAS

The condition for a repeat post-launch is defined as follows:

B_WNAS = 1
IF
the last start was a cold start or a repeated cold start (tmot < K_TI_MK_TMOT_KS)
AND the service life t_motor_stands < KL_TI_MK_WKS_MS_TMOT
AND B_FLAG_WNAS was set
OTHERWISE
B_WNAS = 0.

The repeat after start flag B_FLAG_WNAS (BIT5 in start_st) is set,

IF
the engine is switched off (B_KLA)
AND the engine running time is within the limits when switched off
K_TI_MK_TMIN_WNAS <t_ml <K_TI_MK_TMAX_WNAS bewegt,
OTHERWISE
B_FLAG_WNAS is deleted.

It is then saved in NVRAM.

The restart factor ti_mk_f_nas is only used in the MOTOR RUNNING operating state calculated:

(19) ti_mk_f_nas_word (k) = ti_mk_f_nas_word (k-1)
- (ti_mk_f_nas_word) (k-1)
* ti_mk_tau_nas (k))

(20) ti_mk_f_nas (k) = 1 + ti_mk_f_nas_word (k)

The factor ti_mk_f_nas_word is only calculated in the START operating state and then used as the start value for the exponential function.

1.2.5.1 On hot start

(21) ti_mk_f_nas_word = KL_TI_MK_TAN_NAS

1.2.5.2 No hot start and no repeat cold start

(22) ti_mk_f_nas_word = KL_TI_MK_TMOT_NAS

1.2.5.3 No hot start and repeat cold post-start

(23) ti_mk_f_nas_word = KL_TI_MK_TMOT_NAS
* K_TI_MK_WNAS

1.2.6 CALCULATION OF THE WARM-UP FACTOR

The warm-up factor ti_mk_f_wl is calculated in the 10 msec task.

The warm-up factor is calculated from B_START and at B_ML and when no partially fired operation is active (!B_SKS_TIEINGRIFF; to protect the catalytic converter).

As soon as the lambda control is active, this factor is reduced or increased to 1.0 via a ramp with the gradient K_TI_D_WL (for MSN64: K_TI_MK_D_WL).
Re-triggering can only take place via the B_START status.

Operating status KATHEIZEN:

1.2.6.1 secondary air pump

(24) ti_mk_f_wl = KF_TI_MK_TMOT_TML_SLP_F
* KF_TI_MK_N_WI_SLP_F
+(KF_TI_MK_TMOT_TML_SLP_M
* KF_TI_MK_N_WI_SLP_M)

1.2.6.2 Secondary air pump off

(25) ti_mk_f_wl = KF_TI_MK_TMOT_TML_KAT_F
* KF_TI_MK_N_WI_KAT_F
+(KF_TI_MK_TMOT_TML_KAT_M
* KF_TI_MK_N_WI_KAT_M)

Operating status NO KATHEIZEN:

1.2.6.3 Secondary air pump off and no cat heating

(26) ti_mk_f_wl_long = KF_TI_MK_TMOT_TML_WL
* KF_TI_MK_N_WI_WL

During KATHEIZEN, a correction factor from the characteristic curve KL_TI_MK_TMOT_TAN_DIF, which depends on the temperature difference TMOT-TAN, is added to the calculated factor ti_mk_f_wl.

(27) ti_mk_f_wl =1+(ti_mk_f_wl_long+KL_TI_MK_TMOT_TAN_DIF)

1.2.7 CALCULATION OF THE INDIVIDUAL CYLINDER CORRECTION FACTORS

The correction factor affects the injection time and is determined from an individual characteristic curve over the speed.

(28) ti_f_zyl[i] = KL_TI_N_ZYL [i]

with i = 1, 2, …, n; n = number of cylinders

1.2.8 IDLE SYNCHRONIZATION OFFSET CALCULATION

There is an individual offset for each cylinder, which compensates for the different filling of the individual cylinders with the throttle valve closed via the injection time at low engine speeds.

(29) ti_sync [i] = (K_N_LL_SYNC / n40) * ti_ll_z[i]

with i = 1, 2, …, n; n = number of cylinders

The variables ti_ll_z[i] can be changed both via the application system and via the diagnostic interface and can be saved in NVRAM.

1.2.9 CALCULATION OF THE TORQUE FACTOR

Injection mass factors that influence the engine torque are summarized in one factor and forwarded to the torque manager, chapter "Calculation of lambda efficiency". Only mixture leanings during the warm-up phase are taken into account, factors for mixture enrichment (ti_mk_f_md > 1) are not included.

(30) ti_mk_f_md = ti_mk_f_wl * ti_mk_f_nas
* ((ti_mk_f_kats1 + ti_mk_f_kats2) / 2)

1.3 SEQUENTIAL INJECTION TIME

1.3.1 CALCULATION OF FUEL MASS AND INJECTION TIME

The air mass per cylinder and work cycle ml_zyl is calculated from the product of ml_soll_korr_eff[i] and the cylinder displacement. ml_soll_korr_eff[i] is the corrected air mass per working cycle and cylinder related to the cylinder displacement. ml_soll_korr_eff[i] is specified in [mg/l*ASP]. Since ml_zyl is only used as an intermediate variable and is directly proportional to ml_soll_korr_eff[i] via the cylinder stroke volume, the variable is calculated segment-synchronously net and can also be viewed via an application system, but is not saved individually for each cylinder.

The cylinder-selective injection mass is formed from the quotient of the air mass and the stoichiometric air-fuel ratio.

The relationship between injected fuel mass and injection time is as follows:

(1) mk_zyl [i] = K_TI_EV_QSTAT * ti [i]

With:

mk_zyl[i]: cylinder-selective fuel mass [mg]
ti[i]: effective, cylinder-selective injection time [ms]
K_TI_EV_QSTAT: Factor from injector characteristic curve [mg/ms](pressure-dependent)

From Eq. (1) follows:

(2) ti [i] = mk_zyl [i] / K_TI_EV_QSTAT

1.3.2 OPERATING STATUS START

1.3.2.1 Fuel mass in START

If the START condition is met, the cylinder-selective injection mass is:

(3) ml_zyl = ml_soll_korr_eff[i] * K_RF_HUBVOLUMEN / cfg_cylinder number

(4) mk_zyl [i] = (ml_zyl / K_TI_L_STOECH) Starter injection (zyl.selective)
* ti_mk_f_ga Basic adjustment factor
* ti_mk_f_start Starteinspritzfaktor
* ti_mk_start_f_p_umg ambient pressure-dependent factor

There is no fuel balancing in the injection operating mode transitions module in the START operating mode.

1.3.2.2 Injection time in START

In principle, after the calculation of mk_zyl, the tiueb module is called up to balance the fuel masses, but it does not make any contribution in the Start operating state, so that using Eq. (2) the corrected, cylinder-selective injection time at the start results in:

(5) ti_ [i] = ((mk_zyl [i] / K_TI_EV_QSTAT
* ti_f_adapt[j]) + Adaptation factor (bank select)
ti_offset_adapt[j]) + Adaptation offset (bank select)
ti_sync[i] Idle Sync Offsets (cylinder individual)

Note for software developers: In the software, for the operating modes

Starting and engine running uses the same formula. However, the factor ti_f_zyl is always 1.0 at the start, because it is only interpolated from characteristic curves in the full-load operating state, which cannot occur at the same time as the start operating state.

1.3.3 OPERATING CONDITION MOTOR RUNNING

1.3.3.1 Fuel mass with ENGINE RUNNING

If the ENGINE RUNNING operating mode is active, the cylinder-selective injection mass is calculated as follows:

(6) ml_zyl = ml_soll_korr_eff[i] * K_RF_HUBVOLUMEN / cfg_cylinder number

(7) mk_zyl [i] = (ml_zyl / K_TI_L_STOECH) Ground injection mass (zyl.selective)
* ti_mk_f_ga Basic adjustment factor
* ti_mk_f_stat Stationary factor
* ti_mk_f_nas post-launch factor
* ti_f_mk_wl Warmlauffaktor
* ba_f_ti acceleration enrichment
* ti_mk_f_we Reset factor
* ti_mk_f_sks factor re. security concept (K_TI_MK_SKS)
* ti_mk_f_kats [j] CAT protection factor (bank-selective)

The fuel mass calculated here is now used for fuel balancing in the injection operating mode transitions module. In SES operation, the VL fuel mass is required for balancing in addition to the currently required fuel mass. The VL fuel mass is determined as follows:

If an operating mode transition from FES to SES is detected, the VL target air mass at the current engine speed follows with the corrected, maximum indicated work wi_max (torque manager module) from the map KF_ML_SOLL_BAS_5 (target air mass SES+4V). The resulting VL setpoint air mass still has to be related to the current ambient conditions.

The VL fuel mass is then calculated analogously to Eq. (6) and (7):

(8) ml_vl_zyl = ml_soll_vl_korr_eff[i] * K_RF_LIFT VOLUME/cfg_cylinder number
(9) mk_vl_zyl [i] = (ml_vl_zyl / K_TI_L_STOECH) VL-Injector (zyl.selective)
* ti_mk_f_ga Basic adjustment factor
* ti_mk_f_stat Stationary factor
* ti_mk_f_nas post-launch factor
* ti_mk_f_wl Warmlauffaktor
* ba_f_ti acceleration enrichment
* ti_mk_f_we Reset factor
* ti_mk_f_sks factor re. security concept (K_TI_MK_SKS)
* ti_mk_f_kats [j] CAT protection factor (bank-selective)

Note: The VL fuel mass is only to be calculated in the SES operating mode.

Ml_vl_zyl is just an auxiliary variable that is not visible in the application system.

1.3.3.2 Injection time with ENGINE RUNNING

After mk_zyl[i] has been calculated, the tiueb module is called for fuel balancing. The passed parameters are mk_zyl[i] and mk_vl_zyl[i]. The tiueb module supplies a corrected fuel mass mk_korr, which in turn is the input variable for calculating the injection time.

Using Eq. (2) and after the fuel mass balancing, the corrected, cylinder-selective injection time in the ENGINE RUNNING operating mode is:

(10) ti [i] = (mk_korr / K_TI_EV_QSTAT	
* ti_f_adapt[j])	Adaptation factor (bank select)
+ ti_offset_adapt[j])	Adaptation offset (bank select)
* ti_f_zyl[i]) +	cylinder-specific factor
ti_sync[i]	Idle Sync Offsets (cylinder individual)

mk_korr designates the fuel mass resulting from the balance calculation in the current segment.

1.3.4 INJECTION TIME LIMITATION AND UBATT CORRECTION

In general:
The injection time is limited below to K_TI_MIN and above to K_TI_MAX.

Then the vehicle electrical system voltage correction offset ti_ub from the characteristic KL_TI_UB included and the TPU values determined for total injection time:

(11) ti_eff [i] = ti [i] + ti_ub

As an aid to the application, the variables ti_eff_out[i] are calculated in a 10ms grid, which are set to zero for injection suppressions, otherwise but agree with ti_eff[I].

1.4 FUNCTIONAL SCREEN

(to be defined !)

1.5 APPLICATION NOTES

(to be defined !)

1.6 CYLINDER SKIP AND CYLINDER SHOW

1.6.1 Diffusion during push-off

If the fuel cut-off B_SA condition is met, all cylinders are suppressed. For this purpose, the injection pulses started are injected and also ignited; Only then are all further injection pulses suppressed, ie every 90 °KW or 120 °KW (at the end of injection) this cylinder is blocked.

1.6.2. Fade in after fuel cut-off

After all cylinders were blanked out, the intake manifold dries up. In order to build up the evaporated intake manifold wall film again when reinserting, you have to supply more fuel than normal.

The restart factor ti_mk_f_we compensates for this additional fuel requirement.

It is calculated as follows:

ti_mk_f_we = 1 + (ti_f_we_off * ti_f_we_ign)

The factor ti_f_we_off depends on how long the fuel cutoff was active. It is calculated from two characteristics over time in SA, with one characteristic for hard and one for soft recovery (KL_TI_WE_OFF_S or KL_TI_WE_OFF_H).

The factor ti_f_we_ign depends on the number of ignitions since reinstatement. This factor is limited to 1.0 via the number of ignitions. It is calculated from two characteristic curves based on the number of ignitions, with one characteristic curve for hard and one for soft reinstatement (KL_TI_WE_IGN_S or KL_TI_WE_IGN_H).

The ignition counter ti_we_ign counts the number of ignitions since reinstatement, regardless of whether it is hard or soft reinstatement.

A cylinder is released again every 90 °CA or 120 °CA (at the fictitious end of injection).

1.7 LOADING THE INJECTION TIME INTO *THE TIME PROCESSOR UNIT*

If the condition for a pre-spray B_VSP is met, it is output.

If the condition for the sequential injection B_SSP is met, the TPU parameters for the injection times are updated in the 90 or 120 °CA task and the TPU parameters for the injection ends are updated every 720 °CA.

1.8 INJECTIONS

The end of injection is calculated relative to the inlet valve closing, ie 200 °CA means the end of injection is 200 °CA before the inlet valve closes.

There are different operating states for the final injection value
each a constant. Currently there are:

K_TI_ENDE_MAN, K_TI_ENDE_START, K_TI_ENDE_VL,
KL_TI_ENDE_0 (up to 5), K_TI_ENDE_11.
The filtering mechanism implemented in the MSSxx has been removed.

2 MODULE DATA

The function is **calculated segment-synchronously** in the master.

	Store	background	1ms	10ms	20ms	100ms	1s
Task	x						

variables

Variable	Initialization unit 0p mg/Asp air mass per		area quant. implement 0p-1638p 1/40p			page
ml_zyl	cylinder and working cycle				your	
mk_zyl[i]	0p mg/Asp Cylinder-selective fuel mass for		0p-131p	0.002p your		
	balancing					
mk_vl_zyl[i]	0p mg/Asp 0p-131p Cylinder-selective VL fuel mass for balancing			0.002	your	
ti_ub	0p 0p-65.53p 0.001p your	ms				
	Vehicle electrical system voltage correction offset for the injection time					
you [i]	0p 0p-65.53p 0.001p your	ms				
	cylinder-selective injection time, without battery voltage correction					
ti_eff [i]	0p	ms	0p-65.53p 0.001p your			
	Effective, cylinder-selective total injection time					
ti_mk_f_ga			0p-2p	1/128	where	
	Basic adjustment factor					
ti_mk_start_f_p_umg	0p		0p-2p	1/128p your		
	Ambient pressure-dependent correction factor for the Start operating mode					
ti_mk_f_start						
	Starting syringe factor					
ti_mk_f_stat			0p-2p	1/128p	where	
	Stationärfaktor					
ti_mk_f_nas			0p-4p	1/1024p your		
	post-start factor					
ti_mk_f_wl			0p-4p	1/1024p your		
	warm-up factor					
ba_f_ti			0p-2p	1/1024p your		
ti_mk_f_we	reinstatement factor		0p-2p	1/128p	where	
ti_mk_f_sks			0p-2p	1/128p	where	
	Factor regarding security concept					
ti_mk_f_kats1,2			0p-4p	1/1024p your		
	Katschutzfaktor Bank1/2					
ti_hide_soll						
	Number of cylinders to hide					
ti_hide_act						
	Number of cylinders actually hidden					
ti_st_soll						
	Status of the target status of the injection (1 = channel active)					
ti_st_psp						
	Status of the actual status of the injection (! = channel active)					
ti_dkba1			0p-65.53p 1/1000 your			
	aftersplash					

ti_isr_count						
	Interrupt counter of PSP interrupts					
ti_st_start						
	Injection status word in START mode					
ti_off_time			0p 268Miop	1/16	the	
	Duration of the fade-out					
ti_zyl_off						
start_st						
	Status word of the START operating state					
ti_fizzonteks			0p-2p	1/128	where	
	Cold start factor over speed					
ti_f_tan_hs			0p-64p	1/1024 your		
	Hot start factor over intake air temperature					
ti_f_tmot_ks			0p-64p	1/1024 your		
	Cold start factor over engine temperature					
ti_f_no_zaehler						
	Bridging factor based on the number of camshaft revolutions when starting					
ti_tz_offset_kats		° KW			where	
	Ignition angle offset for sum of retraction angles for injection correction factor with CAT protection					
ti_kats_st						
	Status for cat protection					
ti_f_kats_steuer1/2			0p-64p	1/1024 your		
	Pre-tax value of the cat protection Bank1/2					
ti_f_kats_regler				1/8192 your		
	Regulator value of cat protection for Bank1/2					
ti_mk_f_f_nas_word				1/32768 your		
	Start value and internal, more precise calculation value for the post-start factor					
ti_mk_nas				1/1024 your		
	post-start factor					
ti_tau_nas				655/(x+1)	your	
	Deceleration time constant for the post-start					

Parameter

appl size	support points	unit	area quant. implement 0p-10p 0.01p			page
K_TI_EV_QSTAT		mg/ms			your	
	Slope factor from the injector characteristic curve					
K_TI_MIN		ms	0p-4p	0.001p your		
	Minimum injection time					
K_TI_MAX		ms	0p-65.53p 0.001		your	
	Maximum injection time					
K_TI_L_STOECH			0p-25p	0.1	where	
	Stoichiometric air-fuel ratio					
K_TI_MK_SKS			0p-2p	0.01	where	
	Leaning factor for partially fired operation					
K_TI_START		ms	0p-65.35p 0.001		your	
	starting amount					
K_TI_MK_NAS			0p-2p	0.01	where	

	Switching threshold for the time constant at NAS					
K_TI_D_WL		%/s	0p-0.63p	10/6553 6	your	
	Warm-up regulation gradient with active lambda control					
K_TI_MK_GA		·	0p-2p	1/128	where	
	Basic adjustment factor					
K_TI_KATS		1 / ° KW	0p-0.01p	10/2621 4	where	
	CAT protection factor					
K_TI_KATS_TABG_EIN		·	0p-2p	0.01	where	
	Switch-on threshold TABG for controller KAT protection					
K_TI_KATS_TABG_SCHN HE		·	0p-2p	1/16	where	
	Threshold TABG for controller KAT protection reinforced					
K_TI_KATS_TABG_AUS		·	0p-2p	0.01	where	
	Switch-off threshold TABG for controller KAT protection					
K_TI_KATS_FAK_SCHNEL L		·	0p-16p	1/16	where	
	Factor for override controller CAT protection					
K_TI_MK_F_KATS_MAX		·	0p-4p	1/1024 your		
	Max Cat Protection Factor					
K_TI_TAU_NAS		·	0p-4p	1/64	where	
	Weighting fact for Tau in NAS					
K_TI_TMIN_WNAS		s	0p-255p	1	where	
	Minimum time for WNAS					
K_TI_TMAX_WNAS		s	0p-255p	1	where	
	Maximum time for WNAS					
K_TI_TMOT_HS		°C	-48p-207p 1		where	
	Tmot threshold for hot start					
K_TI_TMOT_KS		°C	-48p-207p 1		where	
	Tmot threshold for cold start					
K_TI_WKS_B1		·	0p-2p	1/128	where	
	Repeat cold start factor in operating range B1					
K_TI_WKS_B2		·	0p-2p	1/128	where	
	Repeat cold start factor in operating range B2					
K_TI_WNAS		·	0p-1p	1/256	where	
	Repeat Cold After-Start Factor					
K_TENTH_START		° KW	0p-6553p 0.1		where	
	End of injection at start					
K_TENDS_TMOT		°C	-48p-207p 1		where	
	Tmot threshold for Tiende					
K_TIENDE_TMOT_HYS		°C	-48p-207p 1		where	
	Tmot hysteresis for Tiende					
K_TENDE_TAU		ms	0p-5100p 20		where	
	Time constant Tau for Tiende					
K_TENDS_TAU1		ms	0p-5100p 20		where	
	Time constant Tau1 for Tiende					
K_TENDE_N_TAU		1/min	0p-10200p 40		where	
	n-threshold for Tiende Tau					
K_TENDS_TAU2		s	0p-25p	0.1	where	
	Tau for Tiende					
K_T_EKP_ON		ms	0p-65535p 1		your	
	Minimal Einzeit der EKP					
K_TI_MIN		ms	0p-4p	0.0001 your		
	Minimum injection time					
K_TI_MAX		ms	0p-65p	0.0001 your		
	Maximum injection time					
K_TI_NO		1/NW rev Leaning	0p-65535p 1		your	
	factor in part fired operation					
K_TI_PT_KORR_MAX		rpm	0p-10000p 1		your	
	Max. N threshold for PT_KORR factor					

K_TI_AUSS_COUNT		2U	0p-255p	1	where	
	Number of blanks within K_TI_AUSS_BEREICH					
K_TI_AUSS_ZYL			0p-255p	1	where	
	Mask for cylinders to be hidden					
K_N_MAX_VFEHLER		1 / min	0p-10200p 1		your	
	Nmax value with V error					
K_N_LL_SYNC		1/min	0p-10200p 40		where	
	n-threshold for LL synchro					

characteristics

appl size	support points	unit	area quant.	implement 0p-20p		page
KL_TI_UB	In: 6xub Out: 6xti_ub	V ms	0p-65.53p	0.1p 0.001p	your your	
	Injection time correction via UB					
KL_TI_MK_START_F_P_UMG	In: 4xp_umg Out: 4xti_mk_start_f_p_umg	mbar	500p 1150p 0p-2p	3p 0.01p	where where	
	KL for the ambient pressure dependent correction factor					

3 INITIAL CALIBRATION OF THE FUNCTION

Parameter:

K_TI_EV_QSTAT	2.50
K_TI_MIN	0.90
K_TI_MAX	64.00
K_TI_L_STOECH	14.7
K_TI_MK_SKS	0.90

Characteristics:

KL_TI_UB

UB[V]			10	12	14	16
TI_UB [ms]	6 3.88	8 2.06	1.38	1.00	0.76	0.60

KL_TI_MK_START_F_P_UMG

P_UMG [mbar]	701	800	974	1013
TI_MK_START_F_P_UMG [-]	1.00	1.00	1.00	1.00

Project: MSS54

Module: Injection Diagnosis

x. Specify I/O status and actuator control

x.1. control injectors

The injection valve is controlled via the **ti_write** function (unsigned char cylinder number, unsigned char period, unsigned char duty cycle). This function is called from the DS2 software and runs at the **DS2 task level.**

Value ranges:

Cylinder number:	1 .. 8 (cfg_cylindernumber)	
Period duration:	10 .. 100 [msec]	Resolution: 10msec/bit.
duty cycle:	0 .. 100 [%]	Resolution: 1%/bit.

These functions run on the slave.

This function is not executed with **B_ML .** With B_ML, the duty cycle may only be 0%. This is equivalent to switching off the valve.

The function sets the bit (ZYLINDER-1) (= B_EVx_DS2) in **ti_ev_ds2** for the respective cylinder. This prevents the TPU parameters from being loaded by the function software, the TPU channel is configured as LPWM in continuous mode, the period duration and the high time are written to the TPU RAM.

When the diagnosis mode (!B_DIAG) is exited and B_EVx_DS2 is active, the **ti_write_undo** function is called. This function - deletes the B_EVx_DS2

- configures the TPU channels at B_SSP as PSP channel (angle-synchronous injection pulse) and at B_VSP as PWM channel (pre-injection pulse)

The following is returned as a return value:
00: Actuator is properly controlled
01: Control not intended for this cylinder
02: Actuator cannot be controlled because duty cycle is not valid
03: Actuator cannot be controlled because the period is not valid
04: Actuator cannot be controlled because control condition is not met

x.2 Read injection time

The diagnostic software reads the variables **ti1** to **ti8** for "Injection time".
read out (does not yet contain the UBATT correction).

The injection time has a value range from 0 to 65,535 msec (unsigned short) with a resolution of 1 usec/bit.

x.3 Injector Driver Diagnosis

The Harris **HIP82** injector driver diagnoses the following errors:

- open load = interruption
- Short circuit to UB
- Short to ground
- over temperature

The driver status is read out synchronously with the angle (every 720 °KW) and checked for errors in the background task and processed accordingly.

The driver status may only be evaluated if there has been a change in the control signal. This is always the case when there is also new driver status information and the channel is not hidden.

The driver status in the background task is evaluated when

- B_START or B_ML and
- B_SSP (sequ. Injection is active) and - S_KL_15_ROH (KL15 definitely exists) and
- ub > K_TI_UB and
- the injection channel is not hidden and
- the injection channel is not controlled via DS2 and
- there is new driver status information

The driver status is in the variables **ti_ed_ev1** to **ti_ed_ev8.**

After a certain error frequency, the routine enters an error in the error memory.

The following transfer parameters to the **ed_report** are now possible in total:

0x00: no error
0x01: short to battery
0x02: short to ground
0x04: open load
0x08: implausible state

A global diagnosis status variable is also formed for the injector driver => ti_ed_ev_sum (the set bit represents an error for the corresponding
(cylinder+1)

The HIP82 module also detects a shutdown due to **excess temperature** and displays it in the variables **ti_ed_tr1/2 .**

This error evaluation can also be found in the angle-synchronously updated driver status. The evaluation takes place in the background task if

- B_START or B_ML - ub > K_TI_UB - there is new driver status information

and
and

Error	**Effect** of lean	**measure**
open load	exhaust gas rough engine running falsified lambda sensor signal	1. Block the lambda controller in the bank in which the error occurred ÿ LA adaptation blocked; TE blocked; TE adaptation blocked 2. Error memory entry
short circuit +	lean exhaust rough engine running falsified lambda sensor signal Driver switches off automatically	1. Block the lambda controller in the bank in which the error occurred ÿ LA adaptation blocked; TE blocked; TE adaptation blocked 2. Error memory entry
short circuit -	rich exhaust rough engine running falsified lambda sensor signal Injector can continuously be activated ==> cylinder can fill with fuel	1. Block the lambda controller in the bank in which the error occurred ÿ LA adaptation blocked; TE blocked; TE adaptation blocked 2. Error memory entry
overtemperature	lean exhaust rough engine running falsified lambda sensor signal Driver switches off automatically	1. Block the lambda controller in the bank in which the error occurred ÿ LA adaptation blocked; TE blocked; TE adaptation blocked 2. Error memory entry

x. 4 **Specify and read idle synchronization values**

The idle synchronization offset ti_ll_zx can be specified via the DS2 by calling the function **ti_ll_vorg (unsigned** char cylinder, signed short value).
The passed value is checked for its limits (K_TI_LL_MIN and K_TI_LL_MAX) in this function.

The function **ti_ll_read(unsigned** char cylinder) returns the idle synchronization value ti_ll_zx as return value.

ti_ll_zx is a signed short value with a resolution of 1 usec/bit.

PROJECT: MSS54

MODULE : **INJECTION**
INJECTION - TRANSITIONS

AUTHORISATION

AUTHOR (ZS-M-57)____________________ **DATE** __________

APPROVED (ZS-M-57) ____________________ **DATE** __________

APPROVED (EA-E-2) ____________________ **DATE** __________

	Department	date	Name	filename
Author	ZS-M-57	02.08.04	Riksén	4.03

x	Neudefinition		feature change		functional error

Description:

1. Balancing the injected and burned fuel masses during operating mode transitions

Reason:

1. In some operating modes, fuel is stored in the intake manifold and is only drawn in during the next working cycle

Current documentation: Chapter 4.03

Previous changes

Version	date	description
	2.8.2003	First version
S360	10.09.04	Variable list revised/ B.Riksén

	NAME	DEPARTMENT	DATE
AUTHOR	E. OTTO /P. SCHMID	EA-E-2	06.03.04
EDITOR	B. RIKSEN	ZS-M-57	

Table of Contents

1 FUNCTIONAL DESCRIPTION

1.1 PHYSICAL BACKGROUND

This module is used to balance the injected and burned fuel masses. This is at
Operating mode transitions necessary because in some operating modes fuel is stored in the intake manifold and is only sucked in during the next working cycle.
The wall film is not taken into account in this module.
The calculation must be individual for each segment or for each cylinder.

1.2 FES (EARLY ADMISSION CLOSES) AND SES (LATE ADMISSION CLOSES)

In the FES operating mode, the amount of air/fuel mixture required is determined by the valve opening times.

At high engine speeds, the FES operating mode is no longer possible due to the minimum valve opening times and a changeover to the SES operating mode takes place.
In the SES operating mode, the full-load quantity is first sucked in, after BDC the quantity that is not required is pushed back into the intake manifold and only then is the intake valve closed. The quantity pushed back into the intake pipe is then available again for the next working cycle. In the stationary case, the stationary fuel mass is injected in the second working cycle, in the transient case, ie when changing operating modes and load changes, the conditions change, so that the burned and the upstream fuel mass must be balanced.

1.2.1 OPERATION WITH FES

In the FES operating mode, no mixture is pushed back into the cylinder, so that balancing of the fuel quantity is also not necessary.

1.2.2 TRANSITION FROM FES TO SES

If the intake closes early (FES), no mixture is pushed back into the intake port, ie the mixture in the intake manifold is zero. (The wall film is not considered.) When operating with SES, the cylinder is completely filled with mixture, homogenized in the cylinder and part of the charge is pushed back into the intake manifold. In order to achieve a stoichiometric mixture, the fuel quantity for full load must be injected at lambda = 1 in the first working cycle and enriched to the desired air ratio via the correction factor. In the following work cycles, it must be ensured that the mixture in the cylinder and the portion that is pushed back have the applied lambda homogeneously.

1.2.3 OPERATION WITH SES (INCL. CYLINDER DEACTIVATION WITH SES)

When operating with SES, the currently required full-load quantity for lambda = 1 and the fuel quantity that has flowed back in the preceding AS are decisive for the injection. The stationary amount of fuel currently required is not taken into account during injection. This is then included in the fuel quantity of the current AS that has flowed back and is therefore only taken into account in the following AS.

There is an individual, saved quantity for each segment or each cylinder.

mk_korr = mk_vollast - mk_gespeichert [tpu_segm_index] corrected injection quantity

mk_speichert [tpu_segm_index] = mk_vollast - mk_stat stored in intake manifold crowd

1.2.4 TRANSITION FROM SES TO FES

During the transition from SES to FES, the amount of fuel stored in the previous AS is in the intake pipe. If this quantity is greater than that required for the AS, the corrected injection quantity is limited to zero and the stored fuel quantity is set to zero. The correction made is unique. If fuel is still stored in the intake manifold after this AS, the next combustion will be too rich.

(This case occurs when there is a transition from SES to FES and the load is less than 0.7, ie less than half full load.)

mk_korr = mk_stat - mk_saved [tpu_segm_index]

If (mk_stat <= mk_saved [tpu_segm_index]) mk_korr = 0

1.3 2V OPERATION

In 2V operation, only one of the two inlet valves is opened alternately. However, for reasons of symmetry, the injected fuel mass is always distributed evenly to both inlet valves, so that only part of the injected quantity can be sucked into the cylinder.
In alternating 2V operation, fuel is alternately stored in front of the inlet valve, which is kept closed, and sucked in in the next working cycle. A balance must be formed from the last and the current AS.

During the transition to 2V operation, an additional quantity (upstream quantity of fuel) must be injected. The additional amount must not be twice the required amount, otherwise the following AS would not have to be injected. The excess quantity is recalculated in each of the following n working cycles. In order to define the engine behavior, a fuel capture rate is defined, which defines the ratio of the inducted to the injected fuel mass. If the degree of capture is greater than 0.5 and less than 1, there is a curtailing behavior of the excess quantity.

If the 2V operating mode is exited, the current injection quantity is reduced by the additional quantity and the additional quantity is then set to zero. The catch degree takes the value 1, which means that the upstream fuel mass in the next AS also becomes zero.

mk_saved [tpu_segm_index] = mk_korr * (1 - kr_fang) before the valve is closed

mk_Zyl = mk_korr * kr_fang + mk_gespeichert [tpu_segm_index] in the cylinder

mk_korr = mk_stat - mk_transition spoken fuel mass

From this it follows by inserting Eq. 3 in Eq. 2 and with the condition: mk_Zyl = mk_stat

mk_Übergang = (mk_stat - mk_gespeichert [tpu_segm_index]) / kr_fang - mk_stat

mk_korr = (mk_stat - mk_saved [tpu_segm_index]) / kr_fang

mk_saved [tpu_segm_index] = mk_korr * (1 - kr_fang)

kr_fang = variable from the characteristic map over load and speed with the dimension [5,5]

1.4 FUEL DEACTIVATION (SA) AND CYLINDER DEACTIVATION (ZAS)

In the SA and ZAS operating modes, the intake valves are kept closed. If fuel was stored in the previous AS, it remains stored in the intake manifold and is available again in the next active AS.

The last amount of fuel stored in front of it must remain stored in the computer and will be taken into account again for the next active AS.

1.5 FUEL BALANCE

In summary, the fuel mass balance is as follows:

if (SES)

mk_korr = ((mk_vollast - mk_gespeichert [tpu_segm_index]) / kr_fang)

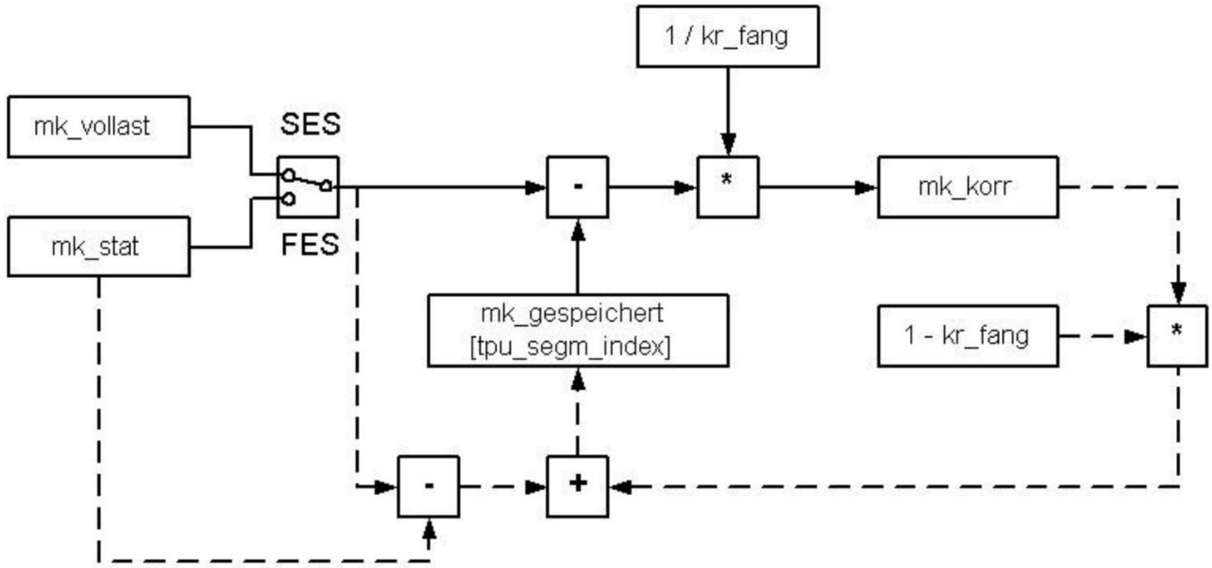
mk_gespeichert [tpu_segm_index] = mk_vollast - mk_stat + (mk_korr * (1.0 - kr_fang))

else if (FES)

if (mk_stat <= mk_saved [tpu_segm_index]) mk_korr = 0

else mk_korr = ((mk_stat - mk_saved [tpu_segm_index]) / kr_fang)

mk_saved [tpu_segm_index] = (mk_korr * (1 - kr_fang))



1.6 APPLICATION NOTES

- application of kr_fang between 0.51 and 1; kr_fang = 1 for 4V and 3V operation

2 MODULE DATA

The function is calculated in the slave.

	Store	background	1ms	10ms	20ms	100ms	1s
Task	x						

variables

Variable	Initialization 0	unit mg /	area quant.	implement		page
mk_stat		asp	0 - 131.07 0.01		uword	
mk_vollast	0	mg / Asp	0 - 131.07 0.01		uword	
mk_saved [tpu_seg_to_datagrave]	0	mg / Asp	0 - 131.07 0.01		uword	
mk_korr	0	mg / Asp	0 - 131.07 0.01		uword	
kr_fang	0			0.001	uword	

Parameter

appl size	support points	unit	area quant.	implement ubytes		page
B_TIUEB			0-1	1		
	0: Tiueb inactive / 1: Tiueb active					

characteristics

appl size	support points	unit	area quant.	implement		page

maps

Applgröße	support points	unit	area quant.	implement 0.	51p to	page
kf_ti_kr_fang	5xn, 5xwi	·	1p		uword	4

Project: MSS54

Module: Acceleration Enrichment

	Department	date	Name	Filename
editor	EE-32	01.04.2013 12:42:00	E. Steger	BA.DOC

1. Enabling conditions for acceleration enrichment

1.1 General release conditions

- The engine is running condition must be **met (B_ML)**
  Acceleration enrichment can be triggered in all engine states
- Torque reduction after START must be completed **(!B_MD_NACHSTART)**
- the speed **threshold K_BA_AKTIV_SCHWELLE** must not be exceeded
- No partially fired operation available **(!B_SKS_TIEINGRIFF)** - to protect the catalytic converter

1.2 Release of a negative / positive acceleration enrichment

Whether a positive or negative acceleration enrichment must be triggered is determined using the "delta air mas - dam" measure.

“Dam” means the change in air mass flow relative to one cylinder.

This value is also normalized via the speed. The calculation takes place in the segment task.

damROH = d_ml_720 / ml_720_min

d_ml_720 = mlx - ml(x-720°KW)
ml_720_min = max [ml, K_HFM_ML_SEG_MIN]

dam = damROH * **$_{nNORM}$**

dam = [-3 .. 3] (nNORM normalized to 1024 rpm)

negative *dam* results when the flap is closed
positive *dam* results when opening the flap

1.2.1 positive acceleration enrichment

- a positive *dam* has occurred
- the change in the air mass flow *dam* exceeds the applicable threshold
  KF_BA_POS_TMOT_N(tmot,n)
- the relative opening **cross-section aq_rel_delta** changes by more than the value
  KL_BA_AQ_DELTA_POS(aq_rel)

If all of these trigger conditions are met, the raw value of the BA factor is determined. For this purpose, the **difference dam_delta** is determined from the actual dam value and the threshold KF_BA_POS_TMOT_N. This difference is the **input value** in the characteristic curve **KL_BA_DAM_POS(dam_delta),** from which the **raw factor ba_fak_roh_signed** is determined.

1.2.2 negative acceleration enrichment

- a negative *dam* has occurred
- the change in air mass flow (amount) *dam* exceeds the applicable threshold **KF_BA_NEG_TMOT_N(tmot,n)**
- the relative opening **cross-section aq_rel_delta** (sign is negative for a neg. BA) changes by more than the value **KL_BA_AQ_DELTA_NEG(aq_rel)**
- Overrun cut-off is not active **(!B_SA).**

If all of these trigger conditions are met, the raw value of the BA factor is determined. For this purpose, the absolute value of the **difference dam_delta** is determined from the actual dam value and the threshold KF_BA_NEG_TMOT_N. This value is the **input** value in the characteristic curve **KL_BA_DAM_NEG(dam_delta),** from which the **raw factor ba_fak_roh_signed** is determined.

2. Calculation of the factor 'ba_f_ti'

If a trigger is detected, a factor is calculated segment-synchronously.

The determined raw factor ba_fak_roh_signed is corrected with

- einem TMOT/TAN abhängigen Faktor **(KF_BA_FAKT_TMOT_TAN(tmot,tan))**

- a speed / RF - factor, depending on whether it is a positive or negative BA.: neg. BA: **KF_BA_FAKT_RF_N_NEG(rf,n)**
  pos. BA: **KF_BA_FAKT_RF_N(rf,n)**

- a recovery factor that compensates for wall film degradation during SA shall be. The input variable in the characteristic curve **KL_BA_FAKT_ZEIT** is the dwell time in overrun cutoff. This factor only comes into play for a time **K_BA_ZEIT_WIEDEREINSETZEN** after reinstatement.

The offset value determined in this way is added to the neutral value "1". This new factor is limited to a minimum **K_BA_FAKT_MIN** and a maximum **K_BA_FAKT_MAX .**

3. Retriggering and start of control

Initial state: BA control is inactive and

- Trigger positive BA => Start of the control POS-BA and takeover of the factor just determined in ba_berech

- Trigger negative BA => start of the control NEG-BA and takeover of the factor just determined in ba_berech

Initial state: POS-BA control is active and

- Trigger positive BA => if the newly determined factor is greater, the value is adopted in ba_berech

- Trigger negative BA => Switch to control NEG-BA and take over the new one Factors in ba_berech

Initial state: Control NEG-BA is active and

-Trigger negative BA => if the newly determined factor is smaller, the value is taken over in ba_berech

- Trigger positive BA => Switch to POS-BA control and accept the new one Factors in ba_berech

Each time the new factor is accepted in ba_berech, the control or
BA factor derating function initialized.

4. Increase or decrease function of the BA factor

The regulation or regulation takes place over 3 levels - in addition, on pos. or neg. BA differentiated:

1. Output of the calculated factor ba_berech in ba_f_ti for the time **KL_BA_IGN_POS/_NEG_TMOT** (for a specific number of ignitions)

ba_f_ti = ba_berech

2. Output of a reduced BA factor in ba_f_ti for the time **KL_BA_IGN_RED_POS/_NEG_TMOT** (for a certain number of ignitions)

ba_f_ti = (ba_berech - 1) * KL_BA_FAKT_RED_TMOT(tmot) + 1

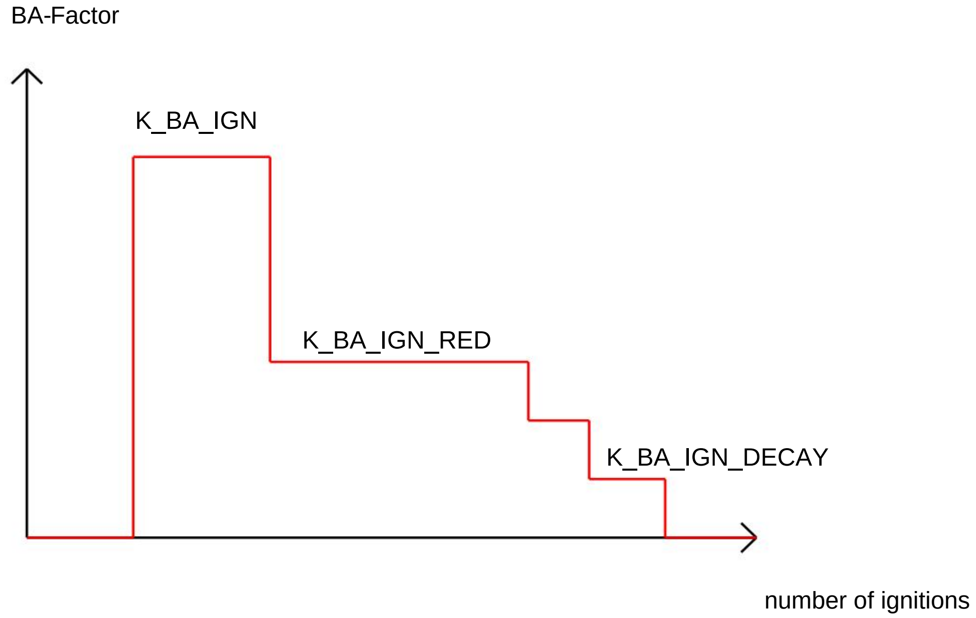
3. Regulation of the BA factor ba_f_ti via a step with a step width of **K_BA_IGN_DECAY_POS/_NEG** (for a specific number of ignitions)

positive Regulation - curtailment to ba_f_ti = 1:

ba_f_tiNEU = ba_f_tiALT - KF_BA_FAKT_RED_POS_TMOT_N(tmot,n)

neg. regulation - regulation to ba_f_ti = 1:

ba_f_tiNEU = ba_f_tiALT + KF_BA_FAKT_RED_NEG_TMOT_N (tmot, n)



4. BA - Abort at idle

In general, acceleration enrichment is triggered in all engine operating states.
However, problems can occur when idling (mixture too rich). Therefore, when entering idle, an active positive BA is aborted.

5. Switching off the lambda controller

When the calculated acceleration enrichment leaves a specific window, the lambda control is switched off.

If ba_f_ti **> K_LA_BA_OFF_POS**

or **ba_f_ti <K_LA_BA_OFF_NEG**

=> **switching off the lambda** controller (la_st_aus - BIT7)

6. Variables and Constants

Name	Type	importance
ba_regel_count	.	Counter for 1st control stage
ba_regel_count_red	.	Counter for 2nd control stage
ba_regel_count_decay	.	Counter for 3rd control stage
ba_berech ba_tmot ba_fakt_time	.	Intermediate value of the BA factor
ba_red_tmot	.	TMOT correction factor
ba_dam_neg_schwelle	.	Correction factor according to SA
ba_dam_pos_schwelle	.	Reduction factor for up/down regulation
ba_aq_delta_neg	.	DAM threshold for neg. BA
ba_aq_delta_pos ba_st ba_f_ti	.	DAM threshold for pos. B.A
	%/segment %/	AQ-REL wave for neg. BA
	segment	AQ-REL threshold for pos.BA
	.	Statusvariable
	.	BA-Factor
K_BA_TIME_REINSERT	TO	Time for map switching
K_BA_AKTIV_SCHWELLE	TO	Speed threshold above the BA is switched off
KL_BA_IGN_POS/_NEG_TMOT	TO	Number of ignitions for f_ti_ba
KL_BA_IGN_RED_POS/_NEG_TMOT K		Number of ignitions for f_ti_ba reduced
K_BA_IGN_DECAY_POS/_NEG	TO	Number of ignitions for f_ti_ba im Abregelvorgang
K_BA_FAKT_MIN	TO	minimum f_ti_ba factor (always positive)
K_BA_FAKT_MAX	TO	maxim f_ti_ba Factor
K_LA_BA_OFF_POS	TO	At Pos BA, from a certain Factor of the LA controller switched off
K_LA_BA_OFF_NEG	TO	At NEG BA, from a certain Factor of the LA controller switched off
KL_BA_AQ_DELTA_NEG	KL=f(aq_rel)	AQ_REL - Threshold for neg. BA
KL_BA_AQ_DELTA_POS	KL=f(aq_rel)	AQ-REL - threshold for pos. B.A
KL_BA_DAM_POS	KL=f(dam_delta) raw factor	for pos. BA dep. by dam_delta
KL_BA_DAM_NEG	KL=f(dam_delta) raw factor	for neg. by dam_delta
KL_BA_FAKT_ZEIT	KL = f (time)	Time since reinstatement
KL_BA_FAKT_RED_TMOT	KL=f(tmot)	Factor as f(tmot) for reduced factor
KF_BA_FAKT_TMOT_TAN	KF=f(tmot,tan)	Faktor als f(tmot,tan)
KF_BA_POS_TMOT_N	KF=f(tmot,n)	DAM threshold for pos. B.A

KF_BA_NEG_TMOT_N	KF=f(tmot,n)	DAM threshold for neg. BA
KF_BA_FAKT_RF_N_NEG	KF=f(rf,n)	weighting factor as f(tmot,n) for neg. BA
KF_BA_FAKT_RF_N	KF=f(rf,n)	weighting factor as f(load,n) for pos. BA
KF_BA_FAKT_RED_NEG_TMOT_N	KF=f(tmot,n)	Red factor as f(tmot,n) for neg. BA
KF_BA_FKAT_RED_POS_TMOT_N	KF=f(tmot,n)	Red. factor as f(tmot,n) for pos. B.A

Statusvariable:

ba_st Status byte for BA

Bit 0: triggering on pos. B.A
Bit 1: Triggering on neg. BA
Bit 2: control pos. B.A
Bit 3: Control neg. BA
Bit 4: ---
Bit 5: ---
Bit 6: ---
Bit 7: ---

MSS54

aftersplash

1. INJECTION RELEASE CONDITIONS

Post-injection is determined segment-synchronously.

The following conditions must be met in order for post-injection to be triggered:

• Full load or partial load •
Max. speed threshold must not be exceeded
(n40 < **K_DKBA_NMAX)**
• min. change of the relative opening cross-section must be given (aq_rel_delta > KL_DKBA_AQ_REL)

• no partially fired operation (!B_SKS_TIEINGRIFF)

A relative charge change over a segment is calculated from the **equivalent map KF_RF_N_AQ_REL** over speed and relative opening cross section:

rf_delta = KF_RF_N_AQ_REL(n, aq_rel) - KF_RF_N_AQ_REL (n,aq_rel_old)

This relative change in charge is another threshold that must be exceeded in order for post-injection to be triggered

=> **rf_delta> KL_DKBA_TRIGGER (n)**

2. REPLACEMENT CALCULATION

If all trigger conditions are met, **rf_delta** is converted into a **dkba_ti_roh :**

=> **dkba_ti_roh = rf_delta * rf_ti_const**

rf_ti_const = K_RF_LIFT VOLUME * K_RF_AIR DENSITY * K_HFM_TI_RATE * 60

2.1. REFILL CALCULATION WITH B_DYN_SOFT

A distinction is now made as to whether the current state is a soft reinstatement.

The post-spray offset to be output is then calculated as follows:

dkba_ti = dkba_ti_roh * dkba_tmot * **KF_DKBA_SOFT_RF_N(rf_roh,n)**

dkba_tmot is calculated from **KL_DKBA_TMOT(tmot) .**

2.2. REPLACEMENT CALCULATION AT B_DYN_HARD

The post-injection offset to be output becomes like a hard restart calculated as follows:

dkba_ti = dkba_ti_roh * dkba_tmot * **KF_DKBA_HARD_RF_N(rf_roh,n)**

dkba_tmot is calculated from **KL_DKBA_TMOT(tmot) .**

An old value is only overwritten if the new post-spray value is larger. dkba_ti is deleted after the after-spray has been triggered in the output function.

3. VARIABLES AND CONSTANTS

rf_delta	-	relative fill change
dkba_tmot	-	TMOT-Faktor
aq_rel_delta	-	AQ_REL threshold
ti_dkba1 dkba_ti	ms	Intermediate splash - MCS representation
dkba_ti_roh	ms/segment	Intermediate splash to be dispensed
	ms/segment	raw score d. intermediate splashes
K_DKBA_NMAX	TO	RPM threshold for post-injection
KL_DKBA_TRIGGER	KL=f(n)	Load threshold for tripping
KL_DKBA_TMOT	KL=f(tmot)	Factor as f(motor temperature)
KL_DKBA_AQ_REL	KL=f(aq_rel)	change of the rel. opening cross-section
KF_RF_N_AQ_REL	KF=f(n,aq_rel) relative	filling gradient
KF_DKBA_SOFT_RF_N	KF=f(rf,n)	Map for soft intermediate splash
KF_DKBA_HARD_RF_N	KF=f(rf,n)	Map for hard intermediate splashes

Project: MSS54

Module: Lambda control

x.2 Lambda control readiness

x.2.1 Lambda probe readiness

Both lambda probes are checked separately for their readiness for regulation.
There are four probe states: - Probe off (cold or defective)
- Probe switch-on monitoring
- Probe on (ready for use)
- Probe switch-off monitoring

You get to the **probe** off state after the reset and from the probe switch-off monitoring state if you were in the probe switch-off monitoring state for the time K_LA_T_AUS.

The probe switch-on monitoring state occurs when the probe voltage is greater than **K_LA_USF** or less than K_LA_USM.

You get into the **probe** on state if you were in the probe switch-on monitoring state for the time K_LA_T_EIN.

The status of **probe** switch-off monitoring occurs when the probe voltage is within the K_LA_USM and K_LA_USF limits.

x.2.2 Switch-on conditions

x.2.2.1 Motor temperature condition

The motor temperature condition is met when:
Im Leerlauf: tmot > K_LA_TMOT_LL
No idle: tmot > K_LA_TMOT
with hysteresis K_LA_TMOT_HYS

x.2.2.2 Probe readiness

Probe readiness is fulfilled when the probe is in the probe on or probe off monitoring status.

x.2.2.3 Application release and DS2 shutdown

The controller is enabled by the constant K_LA_FREIGABE (bit1 for controller 1 and bit2 for controller 2).
The lambda controller can be switched off via the DS2 interface (see diagnosis).

x.2.3 Switch-off conditions

x.2.3.1 Masking

The lambda controller is switched off when one or more cylinders are switched off.
A distinction is made between the two exhaust lines, ie only the control circuit is switched off where the cylinders were blanked out. A suppression can occur in the case of: speed limitation, hard speed limitation, ASC intervention, ASG intervention, overrun fuel cut-off, defective ignition channel, etc

x.2.3.2 Load Threshold

The lambda controller is switched off if the load is longer than K_LA_T_TL above a threshold.

This load threshold is taken from the characteristic curve KL_LA_N over speed.

x.2.3.3 Operating status !MOTOR RUNS or B_VMAX_SOFT

If you are not in the ENGINE RUNNING operating state or if the softer VMAX limitation is in effect, the lambda controller is switched off.

x.2.3.4 With anti-knock enrichment

The lambda controller is switched off when the anti-knock factor ti_f_klops is greater than 1.0.

x.2.3.5 Full load and speed threshold or if the injection time is too short

The lambda controller is switched off when the engine speed is greater than K_LA_N_VL and the operating state is FULL LOAD.

In overrun but not yet with B_SA, the injection time can become so short that the injectors no longer open properly. The controller would then try to enrich and run into the limit. To prevent this, the controller is switched off

if a **tix <** K_LA_TI_MIN

x.2.3.6 Secondary air pump

If the secondary air pump is active or the SLP is activated via the DS interface, the lambda controller is switched off.

x.2.3.7 Idle actuator defective

If the idling actuator is defective, the lambda controller is also switched off in the "idling" operating state.

x.2.3.8 At BA or torque intervention

With acceleration enrichment or with a torque intervention, the lambda controller is switched off if

- the factor ba_f_ti> K_LA_BA_OFF_POS
- the factor ba_f_ti <K_LA_BA_OFF_NEG
- the factor ti_f_smg_x> 1.0
- the factor ti_f_asc_x> 1.0

x.2.3.9 Probe error

Lambda controller 1 or 2 is switched off if there is a sensor error in the relevant bank.

x.2.3.10 Active diagnosis of the secondary air system

Lambda controller 1 or 2 is switched off if the secondary air system is actively diagnosed.

x.2.3.11 Fresh air supply in the exhaust line

When the exhaust line gets fresh air, the lambda controller is switched off.

This can happen if

- the SLP output stage has an error
- the SLP system has an error
- the SLP valve output stage has a fault
- the TE system has a fault
- the TE output stage has an error

x.2.3.12 Nachkath probe short circuit to UB

If the post-cath probe has a short circuit to UB, the lambda controller is switched off.

x.2.3.13 EVT, ZAS and brakes

In the ZAS operating mode, the control circuit is switched off in the event that all cylinders in the control circuit are blanked out, otherwise not. In the braking mode, both control circuits are switched off.

x.3 Lambda probe voltage conditioning

The lambda probe voltage is amplified by a factor of 4.5 with the sensor amplifier LMxxxx and converted by the A/ D converter. The probe voltage is calculated as follows:

$$us = \frac{digit * 5000mV}{1024 * 4,5} = digit * 1,08507$$

The formula in the processor is:

us = (K_LA_US_M * digit) / 1024) - K_LA_US_NP

K_LA_US_M Slope in mv/1024digit
K_LA_US_NP Zero point offset in mV
Both values can be applied.

x.4 lambda probe heater

The lambda probe heating relay is always switched off if you are not in the operating mode ENGINE RUNNING.

In the ENGINE RUNNING operating state, the lambda probe heating relay is switched on with a delay after the end of the start. The delay time is calculated from the KL_LAH_T_EIN characteristic over the engine temperature when exiting the START operating state.

In the ENGINE RUNNING state, the lambda probe heater relay is switched off when the load is greater than a threshold. This threshold is determined from the characteristic KL_LAH_N_AUS over the speed. If the load falls back below this threshold with the hysteresis K_LAH_HYS_AUS, the heating is switched on again.

x.5 Lamb rules

There is one lambda probe for each 3 cylinders and therefore one lambda controller for each of the 3 cylinders.

Cylinders 1, 2 and 3 are controlled by lambda controller 1. Cylinders 4, 5 and 6 are controlled by lambda controller 2.

The lambda controller is a two-point controller of the PITV type, this is a PI controller with a one-sided delay time. A "positive" delay time causes a rich shift and a "negative" delay time causes a lean shift. All three controller parameters (KP, KI, TV) are stored in characteristic diagrams for load and speed.

The two-point behavior comes from the lambda probe, which is a jump probe and therefore only the sign of the control difference can be evaluated.

Therefore, there is also an oscillation of the manipulated variable f_la_controller with an amplitude which is determined by the proportional component la_kp, the integrator slope la_ki and the control path dead time.

Since the dead time depends on the load and engine speed (injection, intake, combustion, exhaust, gas propagation time to the probe, response time of the probe), the controller parameters must also depend on the gas and engine speed.

So that a one-sided lambda shift can be implemented at different operating points, the switchover of the controller is delayed by the time tv. The advantage of this method compared to an asymmetrical P step is that a larger lambda shift can be achieved with the same controller amplitude.

The control formula is:

f_lax = 1.0 + f_la_kp + f_la_ki

mit: f_la_kp = sgn la_kp *

f_la_ki = f_la_ki + (sgn la_ki) *

la_kp is the initial value of the map KF_LA_KP

la_ki is the initial value of the map KF_LA_KI

sgn = -1 if the probe voltage us >= K_LA_UREF ie the exhaust gases are rich.

sgn = +1 if the probe voltage us < K_LA_UREF ie the exhaust gases are lean.

<u>With fat shift:</u>

If there is a jump in the probe voltage from lean to rich, the integrator is stopped for the time tv. If the probe voltage jumps back to lean and the time tv has not yet expired, the integrator is restarted and continues to be integrated until the probe voltage jumps back to rich. Now the time tv continues.

After this time has elapsed, the manipulated variable f_lax jumps by the value f_la_kp = (-1) * la_kp and the integrator integrates from f_la_ki = f_la_ki + (-1) * la_ki .

If the probe voltage now jumps from lean to rich, the manipulated variable jumps again by the value f_la_kp = (+1) * la_kp and the integrator integrates to f_la_ki = f_la_ki + (+1) * la_ki.

The lean shift is analogous to the rich shift.

x.6 Lambdaadaption

x.6.1 principle

The injection quantity is influenced multiplicatively and additively by the adaptation in such a way that the lambda controller corrections become minimal. As a result, the desired lambda is set even when the lambda controller is switched off.
The adaptation compensates for signs of aging and copy control.

The multiplicative factor works with high air flow and high loads.
The additive value works with low air throughput and low speed. It compensates for the leakage air.

Since this is a two-part exhaust system with two lambda probes, the lambda adaptation for the two exhaust lines is also calculated separately.

The adaptation is calculated in the 100msec task.

x.6.2 Adaptionsfreigabe

The adaptation is released when
- the lambda control is active and
- the engine temperature has exceeded the threshold K_LAA_TMOT and
- the intake air temperature is lower than K_LAA_TAN and
- the load is less than a threshold from the characteristic curve KL_LAA_N
- there is no adaptation lock due to the diagnosis and
- the time since the last probe jump is less than K_LAA_T_US and
- the tank ventilation valve is closed.

x.6.3 Adaptation factor: f_ti_a1 and f_ti_a2

The adaptation factor is adapted if
- the air mass greater than K_LAA_ML_SU2 and
- the load is greater than K_LAA_TL_SU2.

The adaptation factor is calculated using the following formula:

f_laax = ((f_lax -1) / K_LAA_TAU2) + f_laax (alt)

The adaptation factor f_laax is limited to K_LAA_FAK_MAX and K_LAA_FAK_MIN.

x.6.4 Adaptation offset: ti_a1 and ti_a2

The adaptation offset is adapted if - the air mass is smaller than K_LAA_ML_SO1 and
- the speed is less than K_LAA_N_SO1.

The adaptation offset is calculated as follows:

laa_regx = low-pass filtered lambda control factor (f_lax) with the time constant K_LAA_TAU.

First, the average control factor deviation of 1.0 is converted into an injection time.

help = (laa_regx - 1) * tl ˙ 2

The injection time determined from this is then integrated (adapted).

laa_offx = (help / K_LAA_TAU1) + laa_offx

The integrator output is minimally and maximally limited.

laa_offx is limited to K_LAA_OFFSET_MAX and K_LAA_OFFSET_MIN.

The adapted value laa_offsetx is then weighted (normalized) via the speed.
Here, the minimum speed n40 is limited to K_LAA_N_NORM_MIN, since the value ti_offset_adaptx could become too large if the speed is too low, e.g. B. with a "dip through the speed" when starting.

The offset that is included in the injection path is:

ti_ax = laa_offx * (K_LAA_N_NORM / n40)

x.7 Diagnose

The lambda controller diagnosis only takes place if
- the lambda controller is active (no mixture-influencing error is present)
- no injection valve is activated via DS2
- there is no probe error

The upper and lower controller stops are checked. The two lambda controllers for the two banks are checked separately.
If the lambda controller receives an extended controller factor due to a suspected short circuit to ground of the probe, the lambda controller diagnosis does not work.

An error (short circuit to plus) is stored if the lambda control factor is limited to the K_LA_FMAX stop for longer than K_LA_T_FMAX. Bit 2 is set in ed_lax.

An error (short circuit to ground) is stored if the lambda control factor is limited to the K_LA_FMIN stop for longer than K_LA_T_FMIN. Bit 1 is set in ed_lax.

x.8 Constants, characteristics, maps, variables

x.8.1 constants

K_LA_RELEASE	Enable constant to turn on and off the Lambda controller and lambda adaptation Bit 0: free Bit 1: Controller 1 enabled Bit 2: Controller 2 enabled Bit 3: free Bit 4: free Bit 5: Adaptation factor enabled Bit 6: Adaptation offset enabled Bit 7: free
K_LA_TMOT	Engine temperature threshold for lambda activation
K_LA_TMOT_LL	Engine temperature threshold for lambda activation in Neutral
K_LA_TMOT_HYS	Engine temperature threshold hysteresis
K_LA_T_TL	Delay time for lambda switch-off under load transgression
K_LA_N_VL	Speed threshold for lambda shutdown at full load
K_LA_UF	Probe voltage for fat threshold at stand-by identifier
K_LA_UM	Probe voltage for lean threshold with readiness detection
K_LA_T_EIN	Switch-on monitoring time for probe readiness detection
K_LA_T_AUS	Switch-off monitoring time for probe readiness recognition
K_LA_US_MAX	maximum probe voltage
K_LA_US_MIN	minimum probe voltage
K_LA_US_TAU	Filter time constant for probe voltage
K_LA_US_NP	Offset for probe voltage processing
K_LA_US_M	Slope of the probe voltage conditioning
K_LA_FMAX	maximum lambda correction factor
K_LA_FMIN	minimum lambda factor
K_LA_T_FMIN	Time threshold for lower controller stop
K_LA_T_FMAX	Time threshold for upper controller stop
K_LA_US_REF	Probe voltage at lambda 1.0
K_LAH_HYS_AUS	Load hysteresis for lambda heating switch-off
K_LAA_TAN	Switch-on threshold of the intake air temperature
K_LAA_TMOT	Motor temperature switch-on threshold
K_LAA_TAU	Time constant for the low pass to smooth the Lambda factors
K_LAA_FAK_MAX	Maximum value of the adaptation factor
K_LAA_FAK_MIN	Minimum value of the adaptation factor
K_LAA_ML_SO1	Upper air mass threshold for the adaptation offset
K_LAA_ML_SU2	lower air mass threshold for the adaptation factor
K_LAA_N_SO1	Upper speed threshold for the adaptation offset
K_LAA_TL_SU2	lower load threshold for the adaptation factor
K_LAA_TAU1	Time constant for the adaptation offset
K_LAA_TAU2	Time constant for the adaptation factor
K_LAA_T_US	Time threshold since last probe jump

K_LAA_OFFSET MAX upper limit of the adaptation offset

K_LAA_OFFSET_MIN lower limit of the adaptation offset
K_LAA_N_NORM normalized speed for weighting of the adaptation offset

K_LAA_N_NORM_MIN minimum speed for weighting of the adaptation offsets

x.8.2 characteristics

KL__LA_N	KL for load threshold for lambda deactivation based on engine speed
KL_LAA_N	KL for load threshold for lambda adaptation via speed

x.8.3 maps

KF_LA_KP	KF for proportional part of the lambda controller
KF_LA_KI	KF for integral part of the lambda controller
KF_LA_TV	KF for the delay time of the lambda controller

x.8.4 variables

st_la	global status byte for Lambda Bit 4: Probe 1 is defective Bit 5: Probe 2 is defective
st_la_e1	Status byte for switch-on conditions of lambda controller 1 Bit 0: Lambda rules 1 active Bit 1: On condition for probe 1 Bit 2: Motor temperature condition Bit 3: Controller release by K_LA_FREIGABE Bit 1 Bit 4: Lambda probe 1 off Bit 5: Lambdasondeneinschaltüberwachung1 Bit 6: Lambda probe 1 on (ready for operation) Bit 7: Lambdasondenausschaltüberwachung 1
st_la_e2	Status byte for switch-on conditions of lambda controller 2 Bit 0: Lambda rules 2 active Bit 1: On condition for probe 2 Bit 2: Motor temperature condition Bit 3: Controller release by K_LA_FREIGABE Bit2 Bit 4: Lambda probe 2 off Bit 5: Lambdasondeneinschaltüberwachung2 Bit 6: Lambdasonde2 on (ready for operation) Bit 7: Lambdasondenausschaltüberwachung 2
st_la_aus	Status byte for switch-off conditions of both lambda rules Bit 0: Cylinders 1,2 and/or 3 are hidden Bit 1: Cylinders 4.5 and/or 6 are hidden Bit 2: Load threshold exceeded Bit 3: START or soft VMAX limitation (B_VMAX_SOFT

	Bit 4: Anti-knock factor or recovery factor are effective Bit 5: full load and n > K_LA_N_VL Bit 6: With secondary air injection Bit 7: free
st_laa	Lambda adaptation status byte Bit 1: Lambda adaptation for controller 1 released Bit 2: Lambda adaptation for controller 2 released Bit 5: Adaptation factor enabled Bit 6: Adaptation offset enabled Bit 7: Adaptation blocked due to diagnostic errors
la_time1	Time at which the load threshold for the Lambdaab circuit was exceeded
la_time2	Time of entry into the lambda probe states SWITCH-ON or SWITCH-OFF MONITORING for probe 1
la_time3	Time of entry into the lambda probe states ON or OFF MONITORING for probe 2
us1	Lambda sensor voltage 1
us2	Lambda sensor voltage 2
la_kp	Proportional component from the map
la_ki	Integral part from the map
la_tv	Delay time from the map
tv1 bzw. tv2	current counter readings of the current deceleration times for lambda controller 1 or 2
st_la_reg1 or 2	Status word of lambda controller 1 or 2
f_la1 or 2	Lambda controller factor (manipulated variable) of lambda controller 1 or 2

f_la_kp1 or 2 Proportional component of the lambda controller factor for lambda controller 1 or 2
f_la_ki1 or 2 integral part of the lambda controller factor for lambda controller 1 or 2

usx_change_time	Time of the last probe jump
f_ti_adapt1 or 2	Total adaptation factor for the injection path
f_laa1 or 2 laa_off1	Adaptation factor 1 or 2
or 2	Adaptation offset 1 or 2 without speed weighting with 32-bit resolution
ti_a1 bzw.2	Adaptation offset 1 or 2 with speed weighting for the injection path
ed_lax	Status variable for lambda controller: Bit 1: lower controller stop Bit 2: upper controller stop Bit 5: error in error filtering Bit 6: Error entered in error memory

Project: MSS54

Module: lambda sensors aging monitoring

	Department	date	Name	Filename
editor	EE-32	01.04.2013 12:46:00		5.02

1. General

This "lambda sensor aging monitoring" function is used to detect aging effects of the sensor before the catalytic converter (VKAT) or a defective probe after the catalytic converter (NKAT) and thus to prevent the emission limit values from being exceeded inadmissibly.

An aging probe can lead to an increase in exhaust gas values both statically (due to a shift in the characteristic curve) and dynamically ("slow" probe).

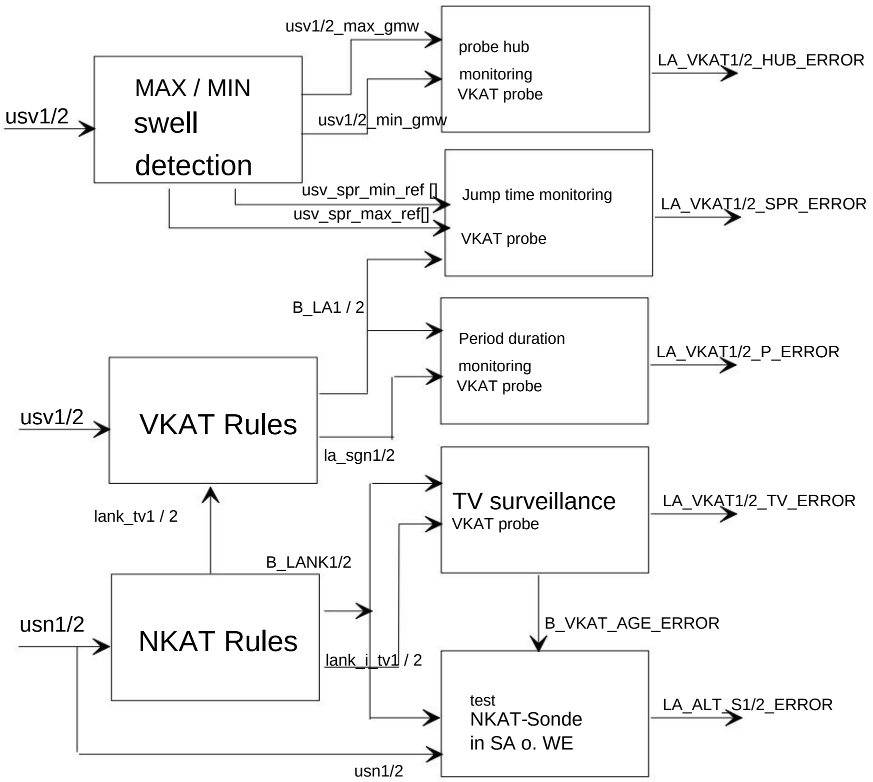
The manipulated variable of the NKAT controller (TV shift) is used to identify and correct a shifted characteristic curve. The following diagnostics are used for monitoring:

- TV surveillance
- Checking of the sensor behind the catalyst during overrun or when reinserted

- Stroke monitoring is used to detect a lambda probe in front of the cat that is too slow
- Periodendauerüberwachung
- Jump time monitoring

of the probe signal is used.

1.1. Lambda sensor aging monitoring overview



1.2. Switch-off conditions of the lambda probe aging monitor

The functions are stopped as soon as one of the following conditions is met:

• Dropout detection **error**
=> B_OUTS_ERROR

• **Throttle potentiometer error**
=> !B_WDK_FEHLERFREI_DPR

• **Probe heating error VKAT or NKAT**
=> B_LSHV1/2_ERROR
=> B_LSHN1/2_ERROR

• **NW - Error**
=> B_TPU_360MODE

• **Error in the tank ventilation system or in the diagnosis**
=> B_TEV_ERROR

• **UBATT - threshold** was undershot
=> ub <= K_ED_UBMIN
=> B_UB_ERRORZ

• **Errors for the VKAT or NKAT probes regarding exceeded Adaptionsfehlerschwellen**
=> LAA1/2_SCHW

• **Secondary air injection with SL diagnosis active or secondary air error is present**
=> B_SLP_ON (is also set here; but is not queried in the diag. - is set via intercepted LA conditions.)
=> B_SLS_CLAMP_ERROR
=> B_SLV_SH_TO_GND

• **Functional Check TEV is active when idle**
=> B_TEFC_LL_CHECK

• **Fuel system diagnostic detects a fault**
=> B_KSD1/2_ERROR

• Stroke monitoring **detects an amplitude that is too small**
=> B_LA_VKAT1/2_HUB_FEHLER

• CAT **protection is active when the tank is empty**
=> B_KATS_MD_RED

All of these general switch-off conditions are combined into a condition **B_LA_ALTER_AUS** (BIT0/1 in la_alter_st), which is used for all lambda probe aging monitoring functions.

2. TV monitoring of the VKAT controller (only for 6-cylinder)

In the 2-probe system, the shift in the VKAT probe characteristic due to aging is overlaid with the TV shift of the NKAT controller. However, if the value of the TV shift leaves a permitted range, then there is a probe error in the VKAT probe.

This diagnosis takes place in a 1s grid.

2.1. Switch-on and switch-off conditions for the aging monitoring of the VKAT sensor

Switch-on condition:

- In order to be able to start this diagnosis at all, the determination of the I component of the NKAT controller must be active.
  => **B_LANK1/2_I**

- Furthermore, it must be ensured that the probe signal is within the filter band.
  is swinging
  => **B_LANK_TAU1 / 2_OK**

- to activate the function , the **BIT0**
  be set. (LA_ALT_TV_RELEASE)

switch-off conditions:

The function is stopped as soon as a

• There is a CAT **conversion** error
An old or defective catalytic converter causes a TV shift, which can lead to a VKAT sensor being incorrectly identified as defective. A defective VKAT probe, on the other hand, blocks the KAT conversion so that a defective catalytic converter cannot be detected at all.

=> is already in the condition !B_LANK1/2_I

• **a general switch** -off condition is present
=>B_LA_AGE_AUS

• **another VKAT probe error** was reported
=> B_LA_AGE_P_ERROR
=> B_LA_AGE_SPR_ERROR

If all switch-on conditions are given and no switch-off condition is active, the Diagnosis enabled => BIT6 / BIT7 in la_alter_st

2.2. Functional definition of the aging monitoring of the VKAT probe

This diagnosis runs continuously in a 1s grid. As soon as the TV shift to be monitored exceeds or falls below the diagnostic thresholds, the diagnostic counter is treated accordingly.

The averaged I component of the trim control is monitored => lank_i_tv_gem[]

If this averaged TV shift exceeds the **maximum threshold K_LA_ALT_TV_MAX,** the diagnosis **counter la_alt_tv_max[]** is incremented by **K_LA_ALT_TV_INC .**

If the TV shift lank_i_tv_gem falls below the **min. threshold K_LA_ALT_TV_MIN,** then the diagnosis **counter la_alt_tv_min[]** is incremented by **K_LA_ALT_TV_INC .**

If **no threshold is exceeded or not** reached , the corresponding diagnosis counter **is decremented by K_LA_ALT_TV_DEC.**

In general, the diagnosis counters la_alt_tv_min/max are limited to ZERO and a maximum of 255.

One speaks of a defective VKAT probe as soon as

la_alt_tv_max []> K_LA_ALT_TV_MAX_COUNT

or

la_alt_tv_min []> K_LA_ALT_TV_MIN_COUNT

is; in this case the status **B_LA_ALTER_TV_FEHLER1/2** (LA_VKAT1/2_TV_FEHLER) set.

As soon as this error has been detected, the following diagnostics are blocked:

- ÿ Periodendauermessung
- ÿ Jump time measurement
- ÿ Monitoring of the NKAT probe (SA/WE check)
- ÿ CAT conversion

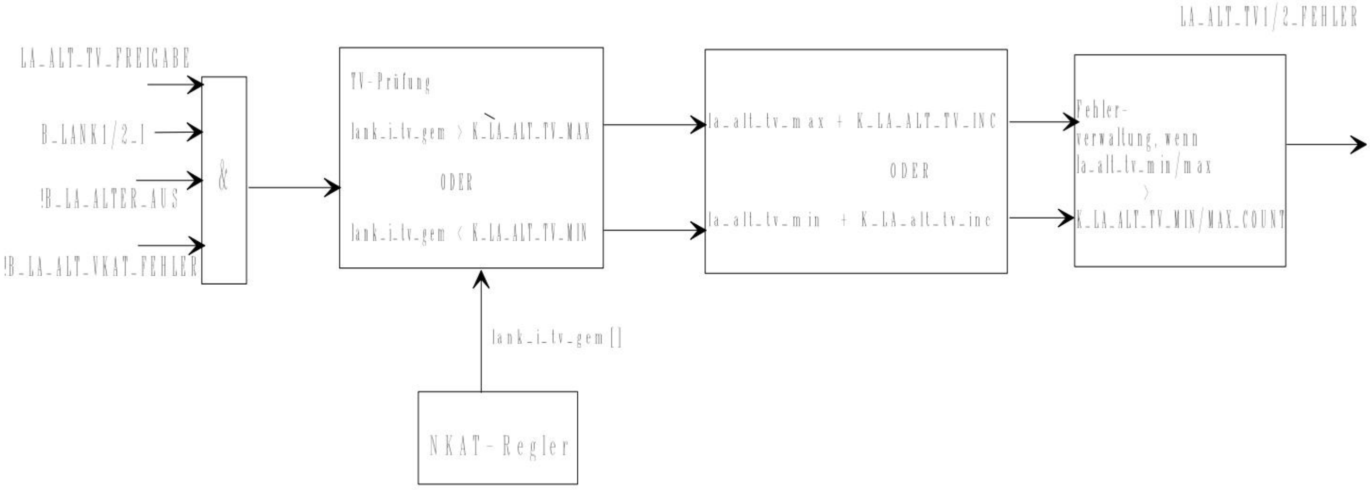
If the min. or max. diagnostic counter threshold is exceeded, either the error of the type **"TV threshold exceeded" (SH_TO_UB)** or **"TV threshold not reached" (SH_TO_GND)** is entered in the error memory with the diagnostic **function ed_report** (the entry appears immediately in the error memory - debouncing counter etc. = 1 - because the debouncing algorithm is in the decrementation of the diagnostic counter.

The error type **"No error present" (NO_FEHLER)** is called when it requires the formation of readyness or when the diagnostic counter is at ZERO for an entered error.
In order to prevent turbo healing when the engine is running again, the diagnosis counter is saved in non-volatile memory if there is an error.

The MIL lamp is activated when the diagnosis detects that the limit value has been exceeded on two consecutive driving cycles (DrCy).

Machine Translated by Google

2.3. Graphic representation of the aging monitoring of the VKAT sensor



3. Period duration monitoring of the VKAT probe signal

With the help of this function, a deterioration in the dynamics of the VKAT lambda probe is detected, which leads to a deterioration in the exhaust gas values.

This period duration measurement takes place every 10ms; directly after the VKAT and NKAT functionalities. The diagnosis itself takes place in a 100ms grid.

3.1. Switch-on conditions for period duration monitoring

The function is enabled when

• **no general switch** -off condition exists
=> !B_LA_AGE_OFF

· **BIT1** is set in the application **constant K_LA_OBD_FREIGABE**
=> LA_ALT_P_RELEASE

• there is **no air mass** error
=>! B_HFM_FEHLER

• **Lambda control VKAT** is active

=> B_LA1 / 2

• the engine **speed is within a certain window** and there is **no N dynamics**
=> **K_LA_ALT_P_N_MIN <n <K_LA_ALT_P_N_MAX**
=> !B_N_DYNAMIK

• The **load is within a specific window** and there are no RF **dynamics**
=> **K_LA_ALT_P_RF_MIN < rf < K_LA_ALT_P_RF_MAX**
=> !B_RF_DYNAMIK_LA

• the exhaust gas temperature is **above a threshold value**
=> **tabg> K_LA_ALT_P_TEMP**

• the **tank** ventilation valve is **closed (!B_TE_SPUEL)** or after opening the valve the time **K_LA_ALT_P_TE_T has** expired.

• there is no trimming due to **clearing** of the CAT
=>! B_LA_KA

• there is no trimming due to **NKAT diagnosis**
=>la_alter_s_tv == 0

• There is no **OBD VKAT probe** error
=> !B_LA_AGE_TV_ERROR
=> !B_LA_ALTER_SPR_ERROR

These conditions are summarized in **B_LA_ALTER_P1/2** (BIT0/1 in la_alt_p_st).

3.2. Periodendauermessung

The period duration is measured between **two FAT-LEAN jumps** of the probe signal (transition from la_sgn: -1 => +1; +blocking time).

First you have to make sure that you are in a **stationary lambda range** (control deviation <= 5%, B_LA1/2_DYNAMIC). Since disturbances can occur after a jump, the period duration measurement is only carried out after the blocking time **K_LA_ALT_P_VERZ_T**
accomplished. If the probe signal is at the "lean" mixture level at the end of the measurement, the period measurement is evaluated as valid. This blocking time is processed after each FAT-LEAN jump. This determined period is **corrected** by the **current TV shift** la_p_tv1/2 (= la_sum_tv1/2) and then, before the actual evaluation, weighted with a weighting factor from the map **KF_LA_ALT_P_FAK_N_RF** => this value can be found in **la_alt_p_mess_of[]**

The mean value **la_alt_p_mess1/2** is formed from these values via a PT1 filter (K_LA_ALT_P_TAU) .

In order to make the period duration monitoring fail-safe, a defined number
K_LA_ALT_P_ANZ_SPR hidden from valid measured periods **from** the start of the function

will; ie the period *counter la_alt_p_anz_spr* is always reset when the switch-on conditions are no longer met.

If the sum of all valid measured periods exceeds the number **K_LA_ALT_P_ANZ_DIAG** *(la_alt_p_anz_diag)* , the average period **la_alt_p_mess1/2** is compared with an upper limit value **K_LA_ALT_P_MAX** and a lower value **K_LA_ALT_P_MIN .**
However, if there is a KAT conversion **error (B_LA_KONV_FEHLER) ,** more stringent thresholds are used - **K_KAT_P_MAX_KONV** and **K_KAT_P_MIN_KONV.**

After the diagnostic time has **elapsed,** either the error of the type **"period duration threshold exceeded" (SH_TO_UB)** or **"period duration threshold not reached" (SH_TO_GND)** or **"no error present" (NO_FEHLER)** is entered in the error memory with the ed_report function - LA_VKAT1 / 2_P_ERROR.

This error entry also occurs only once during engine operation (debouncing counter etc. =1). The MIL lamp is activated when the diagnosis detects that a threshold has been exceeded on two consecutive driving cycles (DrCy).

When deleting the adaptation data or when reading from the FLASH is faulty, the **la_kat_p_mess1/2** period is reset to the INIT value ((K_LA_ALT_P_MAX + K_AL_ALT_P_MIN)/2). Otherwise, **la_kat_p_mess1/2** is initialized with the value **last saved in FLASH** each time it is restarted.

3.2.1. Determination of a valid period

An auxiliary variable la_p_mess_st was introduced to determine a valid period.

From the first measurement after the switch-on conditions have been met:

Set auxiliary variable to starting point la_p_mess_st = 0xFF
1. Increasing the delay time for a FAT -> LEAN jump
la_p_mess_st = la_p_mess_st + 0x80 = 0x7F (= invalid measurement)

2. Checking the delay time in which area you are now
on LEAN: save start time for subsequent period duration measurement; la_p_mess_st
= 0x80 **on BOLD:** this is an invalid range; a measurement cannot be started from here;
la_p_mess_st = 0xFF

3. Increasing the delay time at the next FAT -> LEAN jump;
la_p_mess_st = la_p_mess_st + 0x80

4. Checking the delay time in which area you are now:
on LEAN & la_p_mess_st == 0: valid measurement made; Save start time for subsequent period duration measurement;
la_p_mess_st = 0x80

**on LEAN & (la_p_mess_st == 0x7F ||
la_p_mess_st == 0xFF):**
previous measurement has not been started because one

was in an invalid range;
Start time for measurement can now be saved away, now that you are in a valid area;
la_p_mess_st = 0x80

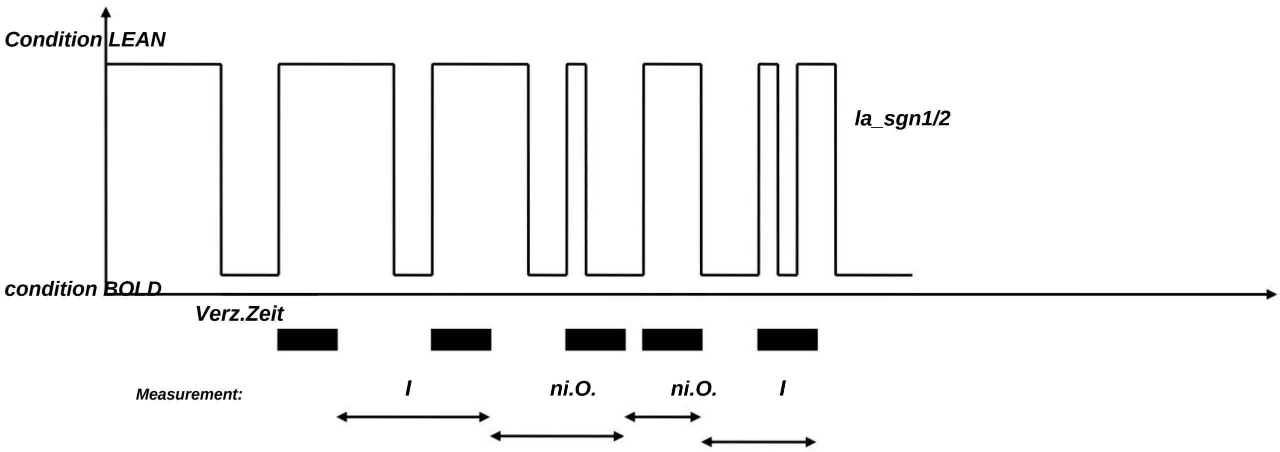
on BOLD: this is an invalid range; a measurement cannot be started from here

if you came from a valid area (la_p_mess_st == 0), then a fault has now occurred; Measurement is aborted => la_p_mess_st = 0xFF

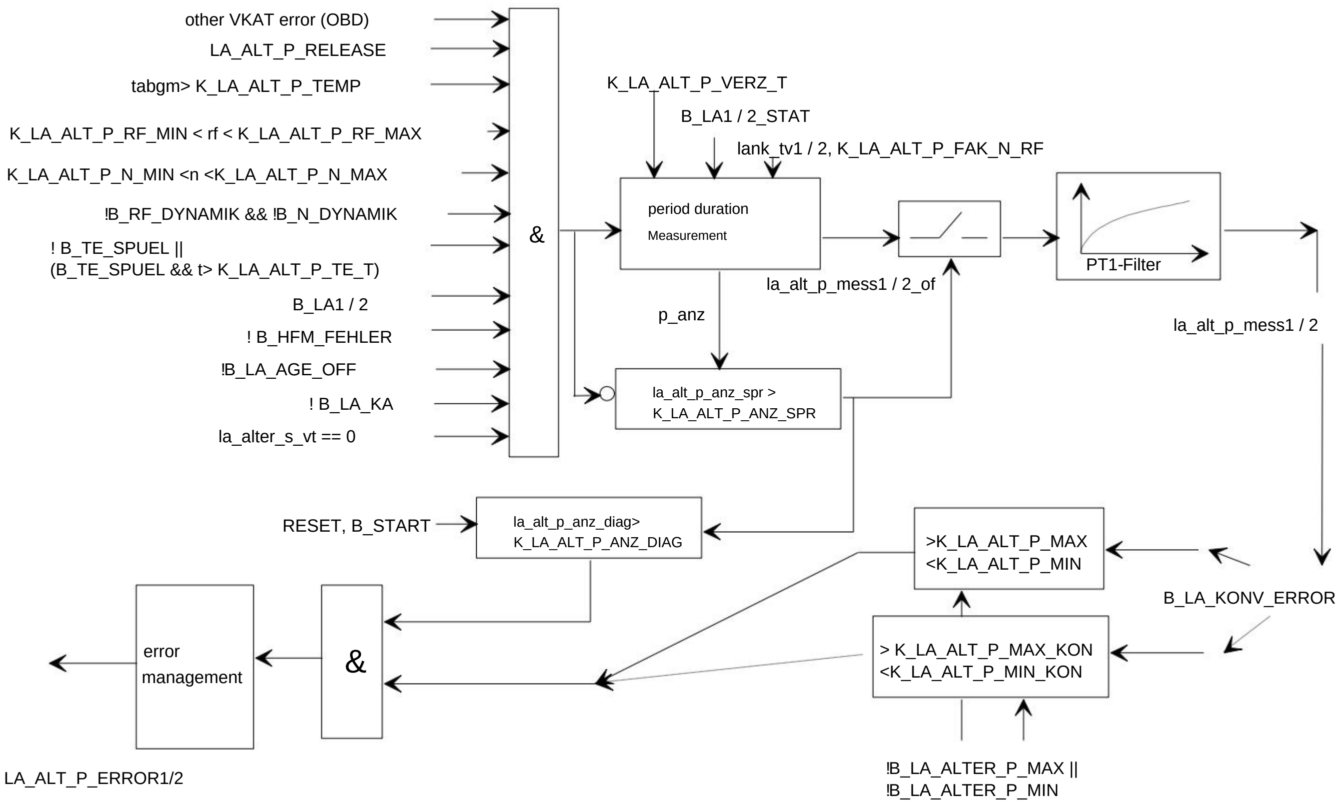
if you came from an invalid area (la_p_mess_st == 0x7F), then the Auxiliary variable not changed (consecutive erroneous periods are thus easy to recognize)

Jump back to point 4

3.2.2. Graphic representation of a measurement



3.3. Graphic representation of the period duration measurement



4. Stroke monitoring of the probe signal VKAT

A diagnosis of the VKAT probe is the monitoring of the probe lift.

For this purpose, the average maximum (usv1/2_max_gemw) and the average minimum (usv1/2_min_gmw) probe voltages are determined.

This measurement takes place every 10ms - the diagnosis is processed continuously

4.1. Switch-on conditions for stroke monitoring

The function is released

• within an RF band

K_LA_USV_GMW_RF_MIN < rf < K_LA_USV_GMW_RF_MAX

The reason for this is that with a very small rf, the probe signal becomes extremely low and thus distorts the minimum mean value. The same happens in the opposite direction, at very high rf - here the upper mean value is distorted.

• the **lambda controller must be active** (B_LA1/2)

4.2. Determination of mean values

Voltage signals that are above or below the limit voltages are used to determine the mean values.

The **limit stresses** are determined as follows and must satisfy the following conditions:

falls

(usv1/2_max_gmw - usv1/2_min_gmw) <= (2 * K_LA_USV_GMW_HYS)

=> usv1/2_limit_upp = usv1/2_limit_low =
(usv1/2_max_gmw + usv1/2_min_gmw)/2

as soon

usv1/2_limit_ob >= usv1/2_limit_bot

=> usv1 / 2_grenz_ob = usv1 / 2_max_gmw + K_LA_USV_GMW_HYS

usv1 / 2_grenz_unt = usv1 / 2_min_gmw - K_LA_USV_GMW_HYS

Machine Translated by Google

The **mean** values are formed via a **PT1 averaging ,** with the **filter constant being K_LA_USV_GMW_TAU .**

The stresses that are above or below the limit stresses are included in the calculation of the mean value.

Condition:

- usv1/2 > usv1/2_limit_ob

  => usv1/2_max_gmw = pt1 (usv1/2, usv1/2_max_gmw, K_LA_USV_GMW_TAU)

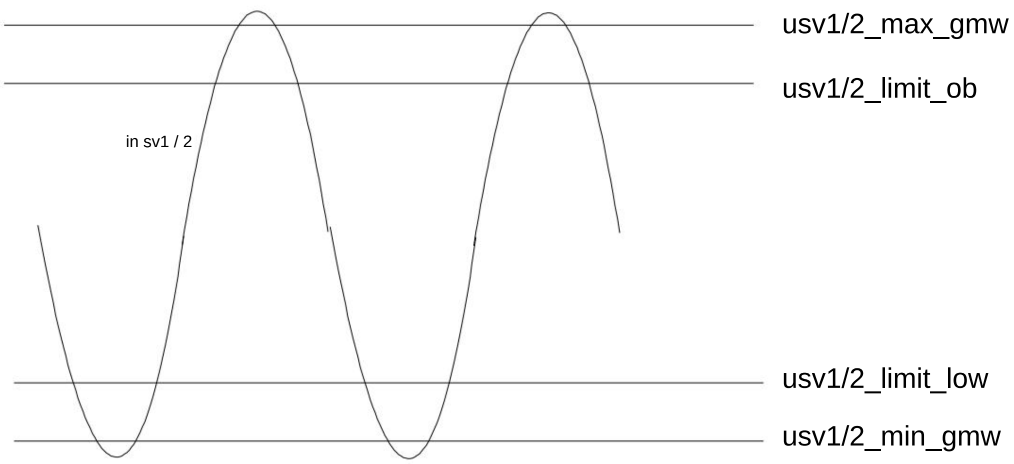
- usv1/2 < usv1/2_limit_low

  => usv1/2_min_gmw = pt1 (usv1/2, usv1/2_min_gmw, K_LA_USV_GMW_TAU)

INITIALIZATION:

The values are reinitialized as follows after a RESET, a new driving cycle or after the error memory has been cleared.

usv1/2_min_gmw = K_LA_USV_GMW_MIN_INI
usv1/2_max_gmw = K_LA_USV_GMW_MAX_INI
usv1/2_limit_ob = K_LA_GRENZ_INI
usv1/2_limit_unt = K_LA_GRENZ_INI



4.3. Hubdiagnose

This diagnosis takes place every 100ms if the following conditions are met.

• the **lambda controller must be active** (B_LA1/2)

• a certain number of P jumps must have taken place
(resets when LA goes inactive)

la_p_spr_count1/2 > K_LA_USV_HUB_P_SPR

probe lift:

la_vkat1/2_hub = usv1/2_max_gmw - usv1/2_min_gmw

A **probe lift error** occurs when the lift falls below a certain threshold

la_vkat1 / 2_hub <K_LA_USV_HUB_DIAG

=> LA_VKAT1/2_HUB_FEHLER

Measures:

In the event of a hub error

• the lambda control of the affected bank stopped because the operational readiness withdrawn

• the adaptation is blocked and reset

• Locked the VKAT and NKAT probe diagnosis

• Locked the KSD diagnostics

• locked the CAT conversion diagnosis

5. Jump time monitoring of the probe signal VKAT

Both probes in front of the CAT are monitored separately for rich and lean switching times.

This can be used to detect a deterioration in the dynamics of the VKAT lambda probe, which leads to a deterioration in the exhaust gas values.

This measurement takes place every 10ms - until the diagnostic time has expired.

5.1. Switch-on conditions for monitoring

The function is enabled when

• **no general switch** -off condition exists
=> !B_LA_AGE_OFF

.

BIT 7 is set in the application **constant K_LA_OBD_FREIGABE**
=> LA_ALT_SPR_RELEASE

• there is **no air mass** error
=>! B_HFM_FEHLER

• **Lambda control VKAT** is active and there are **no LA dynamics**
=> B_LA1 / 2
=> !B_LA1/2_DYNAMIC

• the engine **speed is within a certain window** and there is **no N dynamics**
=> **K_LA_ALT_SPR_N_MIN <n <K_LA_ALT_SPR_N_MAX**
=> !B_N_DYNAMIK

• The **load is within a specific window** and there are no RF **dynamics**
=> **K_LA_ALT_SPR_RF_MIN <rf <K_LA_ALT_SPR_RF_MAX**
=> !B_RF_DYNAMIK_LA

• the **reference thresholds from which the jump time is determined are** calculated
=> B_LA_ALTER_SPR_REF1/2

• the exhaust gas temperature is **above a threshold value**
=> **tabg> K_LA_ALT_SPR_TEMP**

• there is no trimming caused by cleaning out the CAT **and** there is **enough air through** flowed through **the KAT**
=>! B_LA_KA &&! (La_kat_ausr_st & BIT_KA_LANK_ML_SCHW)

• there is no trimming due to **NKAT diagnosis**
=>la_alter_s_tv == 0

• There is no **OBD VKAT probe** error

=> !B_LA_AGE_TV_ERROR
=> !B_LA_AGE_P_ERROR

These conditions are summarized in **B_LA_ALTER_SPR1/2** (BIT0/1 in la_alt_spr_st).

5.2. Determination of the reference thresholds

Relative thresholds are used to determine the switching times of the probe (rich -> lean and lean -> rich). These relative thresholds are 10% and 90% of the signal swing.

The signal swing is made up of the upper signal value usv1/2_max_gmw and the lower signal value usv1/2_min_gmw (see swing monitoring for determination).

The reference thresholds are recalculated every 1s.

Prerequisite: a certain number of **P jumps** after LA active must have expired:

la_p_spr_count1 / 2> K_LA_USV_SPR_P_SPR

Detection:

- **usv_spr_min_ref** [] = usv1 / 2_min_gmw
  + ((usv1/2_max_gmw - usv1/2_min_gmw) * 0.1)

  => **10%** of the signal swing, based on the lower signal value
  => set **BIT2** in **la_alt_spr_st**

- **usv_spr_max_ref** [ ] = usv1/2_min_gmw
  + ((usv1/2_max_gmw - usv1/2_min_gmw) * 0.9)

  => 90% of the signal swing, related to the lower signal value
  => set **BIT3** in **la_alt_spr_st**

5.3. Monitoring for reversal points

In order to detect improper switching of the probe, the probe signals are monitored for reversal points (RICH / LEAN peak) during the measurement of the switching times. If a reversal point is detected, this signal change is not used for diagnosis

FAT peak (fat -> lean - jump):

usv1 / 2 (n)> usv1 / 2 (n-1) + K_LA_ALT_SPR_HYS

Signal increases again by more than K_LA_ALT_SPR_HYS during a signal change to LEAN an.

LEAN peak (lean -> rich - jump):

usv1 / 2 (n) <usv1 / 2 (n-1) - K_LA_ALT_SPR_HYS

Signal decreases again by more than K_LA_ALT_SPR_HYS during a signal change to BOLD.

5.4. Determination of the switching times

5.4.1. Determination of the switching time from RICH to LEAN

The lambda probe signals are sampled in a 10ms grid. As long as the probe signal is greater than the upper reference threshold, the jump time counter is set to zero. As soon as the threshold is undershot, the counter is incremented with each sampling process until the signal undershoots the lower threshold.

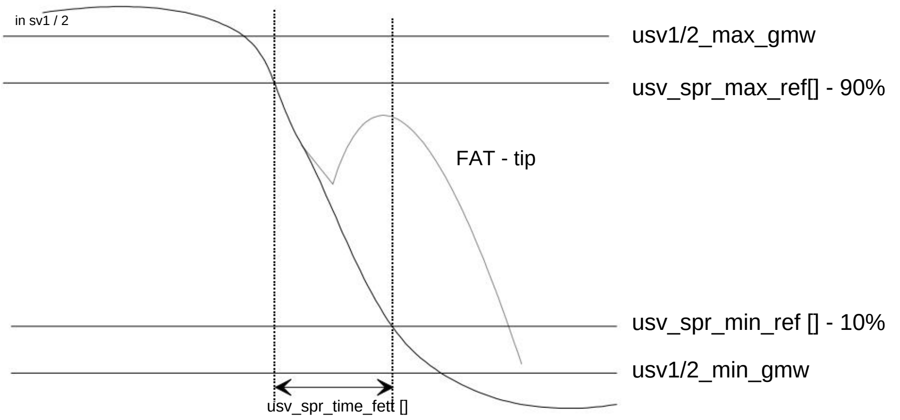
for

usv_spr_max_ref [ ] > usv1/2 > usv_spr_min_ref [ ]

=> usv_spr_time_fett (n) = usv_spr_time_fett (n-1) + 1

In general, **BIT4 / Bank1** or **BIT5 / Bank2** is set in the la_alt_spr_st **status byte** as soon as the probe voltage exceeds the upper reference **voltage usv_spr_max_ref** and is only withdrawn when it falls below the lower reference voltage **usv_spr_min_ref.**

If a fat peak occurs during the determination, the determination of the switching time is aborted and the respective switching time is no longer processed. In this case, BIT4 / Bank1 or BIT5 / Bank2 is withdrawn again.



5.4.2. Determination of the switching time from LEAN to FAT

The lambda probe signals are sampled in a 10ms grid. As long as the probe signal is less than the lower reference threshold, the jump time counter is set to zero. Once the threshold is exceeded, the counter is incremented on each sample until the signal crosses the upper threshold.

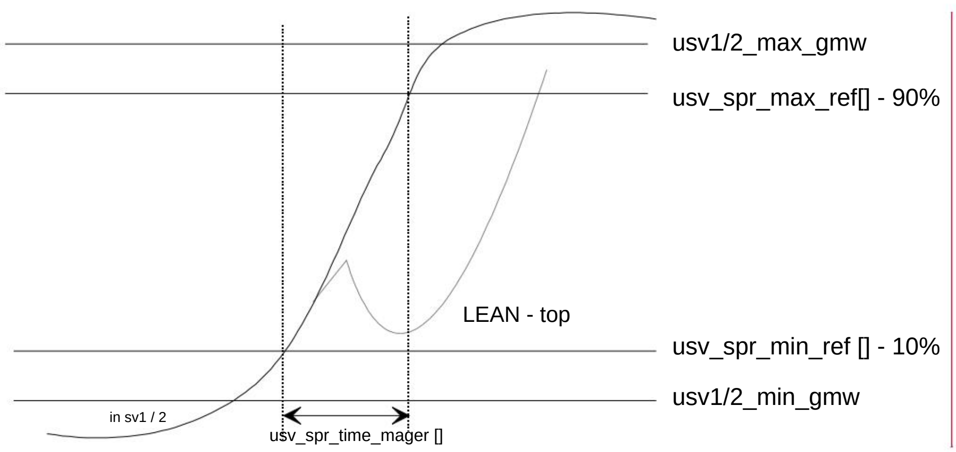
for

usv_spr_min_ref [ ] < usv1/2 < usv_spr_max_ref [ ]

=> usv_spr_time_mager (n) = usv_spr_time_mager (n-1) + 1

In general, **BIT6 / Bank1** or **BIT7 / Bank2** is set in the la_alt_spr_st **status byte** as soon as the probe voltage falls below the lower reference **voltage usv_spr_min_ref** and is only reset when the upper reference voltage usv_spr_max_ref is **exceeded.**

If a lean peak occurs during the determination, the determination of the switching time is aborted and the respective switching time is no longer processed. In this case, BIT6 / Bank1 or BIT7 / Bank2 is withdrawn again.



5.4.3. Averaging the switching times

Since the switching times of the probes vary greatly, an averaging over the entire diagnosis time (K_LA_ALT_SPR_ANZ_FETT / K_LA_ALT_SPR_ANZ_MAGER - measurements) is carried out.

Theoretical jump time - depending on the operating point:

la_alt_spr_m / f_grenz [] = KF_LA_ALT_SPR_MAGER / FETT_GRENZ (n, rf)

Determining the quotient - including the theoretical jump time:

usv_spr_m / f_quot [] = usv_spr_time_mager / fett [] / la_alt_spr_m / f_grenz []

Summation of the quotients:

usv_spr_m/f_quot_sum(n)[] = usv_spr_m/f_quot_sum(n-1)[] + usv_spr_m/f_quot[]

average "jump time":

usv_spr_mager/fett_gem[] = usv_spr_m/f_quot_sum[] / la_alt_spr_anz_m/f[]

=> the result is a **quality feature** and not a time in msec. The actual jump time, which is also output via the scan tool, would result as follows:

actual jump time = figure of merit * theoretical jump time

5.5. jump time diagnosis

The actual diagnosis is made using the usv_spr_mager/fett_gem quality feature:

If the complete diagnosis has expired on both banks, ie

la_alt_spr_anz_m = K_LA_ALT_SPR_ANZ_MAGER
AND la_alt_spr_anz_f = K_LA_ALT_SPR_ANZ_FETT

is checked for exceeding the limit values:

When

usv_spr_mager_gem []> K_LA_ALT_SPR_M_QUOT

OR

usv_spr_fett_gem[] > K_LA_ALT_SPR_F_QUOT

With the **ed_report** function, either the error of the type **"Lean jump time too long" (SH_TO_UB)** or **"FAT jump time too long" (SH_TO_GND)** is entered in the error memory - LA_VKAT1/2_SPR_FEHLER.

This error entry also occurs only once during engine operation (debouncing counter etc. =1). The MIL lamp is activated when the diagnosis detects that a threshold has been exceeded on two consecutive driving cycles (DrCy).

6. Checking the probe behind CAT

This check is performed on the boost or during recovery. In this case, the probe voltage must fall below or exceed a defined voltage threshold.
This diagnosis must be run through once per engine run (either the test according to SA or WE).

The diagnosis is always reopened after a RESET; if only the start is used, all times and air mass quantities ect. reset. However, a diagnosis that has already expired is not restarted.

6.1. Determination of the probe position

Since this check takes place during the overrun or during recovery of the WE, the starting position of the NKAT probe must be checked before the diagnosis.

Verification takes place when

- the engine is running (B_ML)
&&
- you are not in SA (!B_SA).
&&
- an NKAT diagnosis is not currently running (!B_LA_ALTER_DIAG)

The probe voltage usn1/2 is checked to see whether it exceeds the threshold K_LA_ALTER_US_FETT and falls below the maximum threshold K_LA_ALTER_US_FETT.

If **right before SA**

usn1/2 > K_LA_AGE_US_FAT (la_alt_mess_st, BIT2/3)

AND

usn1/2 < K_LA_ALTER_US_FAT_MAX (la_alt_mess_st, BIT6/7)

you are dealing with a **rich mixture** and thus the signal can be monitored **at SA .** If the signal is **below the K_LA_ALTER_US_FETT threshold at this point** in time , the signal can be checked **when** it resumes .

6.2. Conditions for the diagnostic window

The check is carried out if you are in the defined *diagnosis window* during the *entire diagnosis period :*

- The function must be activated via the application constant **K_LA_OBD_FREIGABE, BIT2** being
- A certain speed range must be maintained
  K_LA_ALTER_S_NMIN < n < K_LA_ALTER_S_NMAX
- the engine (time after START) must have been **running longer than K_LA_ALTER_S_TML**
- the KAT temperature **tkatm** must reach a certain threshold **K_LANK_TKAT_SCHW exceeded** (lank_st_ein1/2, BIT_LANK_TKAT_SCHW)
- There is **no CAT protection when the tank** is empty
  ÿ !B_KATS_MD_RED
- there are **no dropouts**
  ÿ !B_OUTS_ERROR
- There are **no secondary air** faults
  ÿ !B_SLS_CLAMP_ERROR
  ÿ !B_SLV_SH_TO_GND
- There is **no other VKAT probe** error
  ÿ !B_LA_VKAT1/2_P/SPR_FEHLER
  ÿ !B_LA_VKAT1/2_HUB_FEHLER
  ÿ !B_LA_VKAT1/2_TV_FEHLER
- There is **no KSD** error
  ÿ !B_KSD1/2_ERROR

6.3. Defined initial state for the test

For the test, both at SA and at reinstatement must

• you are in the boost state for longer **than** a **time K_LA_ALTER_S_SA_T**

&&

• an applicable **air volume K_LA_ALTER_S_ML must have flowed** through the CAT

=> so you have described a defined status for the diagnosis.

However, there is **an exception** to the test in the overrun - if **the test is positive before the defined initial state is reached** (as described below), **the diagnosis is not aborted but recognized as having been carried out.** The aim is to carry out a positive diagnosis as quickly as possible, since a diagnosis in WE is relatively critical.

6.4. Examination in progress

If all test conditions are met, ie

• You are in the diagnosis window
• the probe position was in the fat before SA
• a defined SA has expired (with one exception)
• the NKAT probe is ready (B_LANK_SONDE_BEREIT)
• No electrical probe error and heating error is present (!B_LASV/N_FEHLER, !B_LSHV/N_FEHLER)

then the probe voltage usn1/2 is compared with a threshold K_LA_ALTER_S_SA_US, which must be fallen below at SA.

When

usn1 / 2> K_LA_ALTER_S_SA_US

then it can be assumed that the lambda probe NKAT has aged so much that it either takes too long to fall below this threshold (ie probe too slowly) or it can no longer follow the mixture (gets stuck).

If the probe is recognized as OK, ie the probe signal falls below the threshold - even during the defined SA *(usn1/2 < K_LA_ALTER_S_SA_US),* then the diagnosis ended for this driving cycle and the error counter la_alter_s_count1/2 reset.

To avoid error detections, an error is only entered with the **ed_report** function if the error counter **la_alter_s_count1/2** is greater than **K_LA_ALTER_S_COUNT** (the error counter is always incremented when a check, regardless of whether it is SA or WE, is not recognized as valid). In this case, the error of the type **"Voltage too rich in SA" (OPENLOAD)** is entered in the error location **LA_NKAT1/2_S_FEHLER .**

6.5. Check on reinsertion

If all test conditions are met, ie

- You are in the diagnosis window
- the probe position was lean in front of SA
- only after the defined SA does one come to WE
- the NKAT probe is ready (B_LANK_SONDE_BEREIT)
- no el. probe error and heating error is present

then the probe voltage usn1/2 is compared with a threshold K_LA_ALTER_US_FETT, which must be exceeded during WE.

As soon as the voltage during this WE diagnosis

usn1/2 > K_LA_AGE_US_FAT

then it is assumed that the probe is OK. The diagnosis is ended for this driving cycle, the error counters la_alter_s_count1/2 are reset and also the
Enrichment measures regarding the diagnosis withdrawn (explanation follows).

If during the waiting time **K_LA_ALTER_S_WE_T** (raised during the transition to WE) the probe voltage has not exceeded the diagnosis threshold, an error is not entered immediately, but an additional enrichment **la_alter_s_tv1/2** (is added to la_sum_tv1/2) from the characteristic curve **KL_LA_ALTER_S_TV** (depending on the air mass) determined. This enrichment works for a time **K_LA_ALTER_S_TV_T;** if KAT clearing is active, this is aborted.

The air throughput is also checked so that a clear diagnosis is possible within the enrichment phase. Only when sufficient exhaust gas has flowed through the KAT **(la_alt_s_we_ml > K_LA_ALTER_S_WE_ML)** and the probe has still not exceeded the diagnostic threshold (despite additional enrichment) is it recognized as defective.
Otherwise, the diagnosis is aborted after the times have elapsed.

Interruption of the WE diagnosis:

In general, a WE diagnosis is aborted by an SA phase. Now, however, there is a special situation: ***SA can be recognized during switching operations*** (depending on how SA is applied) .

This detection of SA during switching operations interrupts the WE diagnosis. This can lead to a defective probe not being recognized in a diagnostic cycle, since the WE part is never completed. To counteract this, the WE diagnosis for SA phases is less than a best. Time only stopped and not interrupted.

Stop WE diagnosis (all values are frozen) if:

la_alt_s_sa_we <K_LA_ALTER_S_SA_WE_T

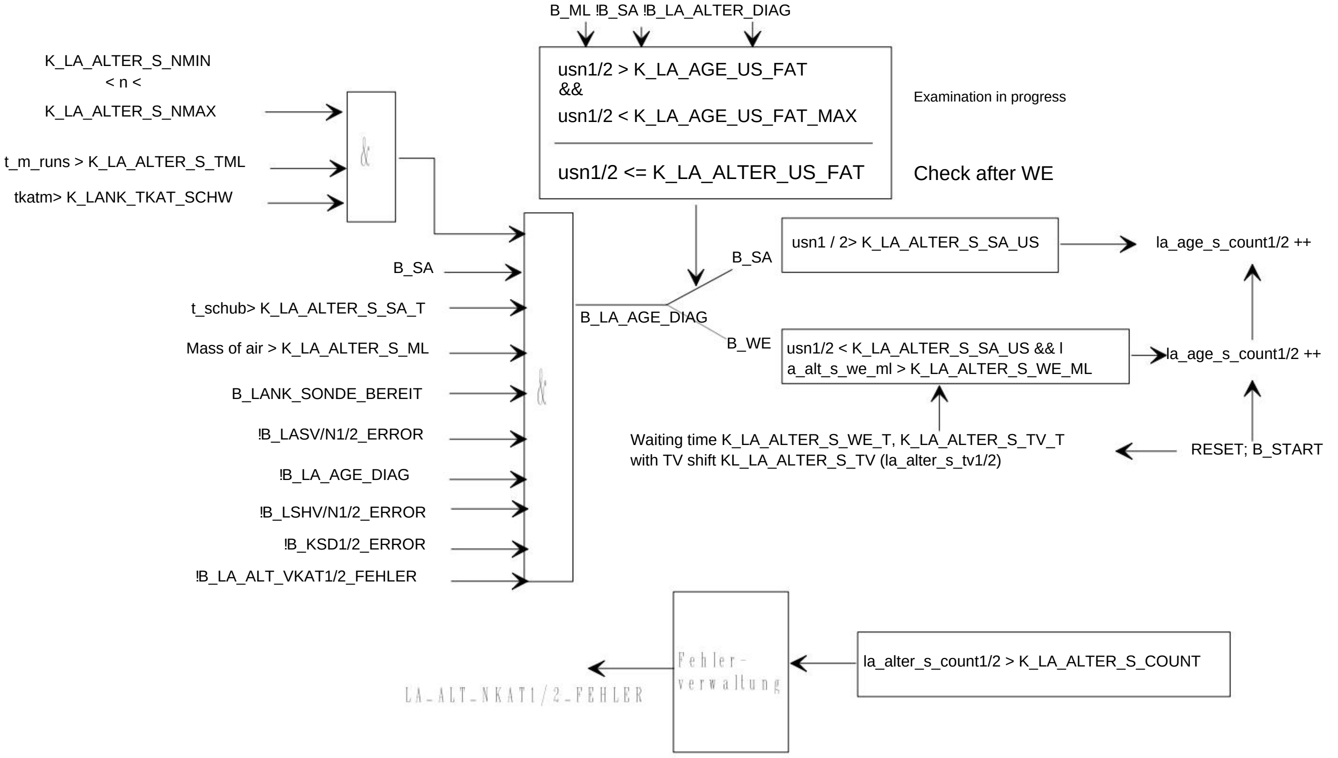
otherwise the WE diagnosis is aborted and the SA path of the diagnosis is run through.

la_alt_s_sa_we: Time from detection of the SA state

In order to avoid error detections, an error is only entered with the **ed_report** function if the error counter **la_alter_s_count1/2** is greater than **K_LA_ALTER_S_COUNT**
(the error counter is always incremented when a check, regardless of whether SA or WE is not recognized as valid). In this case, the error of the type **"Voltage too lean after WE" (IMPLAUSIBLE)** error memory is entered in the error location LA_NKAT1/2_S_FEHLER.

This error entry only occurs once during engine operation (debouncing counter etc. =1).
The MIL lamp is activated when the diagnosis detects that a threshold has been exceeded on two consecutive driving cycles (DrCy).

6.6. Graphical representation



7. Variables and Constants

TV monitoring of the VKAT controller: la_alter_st:

Bit position	la_alter_st
Bit0	B_LA_ALTER_AUS1- general. Switch-off condition Bank1
Bit1	B_LA_ALTER_AUS2- general. switch-off condition Bank2
Bit2	B_LA_ALT_TV_MAX1 - max. threshold exceeded
Bit3	B_LA_ALT_TV_MAX2 - max. threshold exceeded
Bit4	B_LA_ALT_TV_MIN1 - Min. threshold undershot
Bit5	B_LA_ALT_TV_MIN2 - Min. threshold undershot
Bit6	B_LA_ALT_TV_AKTIV1 - TV diagnosis VKAT1 is running
Bit7	B_LA_ALT_TV_AKTIV2 - TV diagnosis VKAT2 running

Period monitoring of the VKAT probe signal: la_alt_p_st:

Bit position	la_alt_p_st
Bit0	B_LA_ALTER_P1 - diagnostic conditions met - VKAT1
Bit1	B_LA_ALTER_P2 - diagnostic conditions met - VKAT2
Bit2	blocking time after FAT-LEAN prog. for VKAT1 has expired
Bit3	Blocking time after FAT-LEAN-Spr. for VKAT2 has expired
Bit4	B_LA_ALTER_P_MAX1 - Period.duration too large - VKAT1
Bit5	B_LA_ALTER_P_MAX2 - Period.duration too large - VKAT2
Bit6	B_LA_ALTER_P_MIN1 - period. duration too small - VKAT1
Bit7	B_LA_ALTER_P_MIN2 - period. duration too small - VKAT2

Jump time monitoring of the VKAT probe signal: la_alt_spr_st:

Bit position	la_alt_spr_st
Bit0	B_LA_ALTER_SPR1 - Diagnostic conditions met - VKAT1
Bit1	B_LA_ALTER_SPR2 - Diagnostic conditions met - VKAT2
Bit2	Reference thresholds for VKAT1 are determined
Bit3	Reference thresholds for VKAT2 are determined
Bit4	Jump time determination FAT -> LEAN takes place - VKAT1
Bit5	Jump time determination FAT -> LEAN takes place - VKAT2
Bit6	Determination of jump times LEAN->FAT takes place - VKAT1
Bit7	Determination of jump times LEAN->FAT takes place - VKAT2

Checking the NKAT probe : la_alt_nkat_st:

Bit-Stelle la_alt_nkat_st

Bit0 B_LA_ALTER_S_SA_BED1- Diag. Bank1 is performed Bit1
B_LA_ALTER_S_SA_BED2- Diag. Bank2 is done
Bit2 B_LA_ALTER_SA_DIAG1- Diag. Nach SA aktiv - Bank1
Bit3 B_LA_ALTER_SA_DIAG2- Diag. Active after SA - Bank2
Bit4 B_LA_ALTER_SA_PHASE - def. SA phase is reached
Bit5 B_LA_ALTER_WE_DIAG - Diagnosis active at WE
Bit6 B_LA_ALTER_WE_TIME - waiting time without further enrichment has expired
Bit7 B_LA_ALTER_WE_TV_TIME - waiting time with additional enrichment has expired

Checking the NKAT probe : la_alt_mess_st

Bit-Stelle la_alt_nkat_st

Bit0 RICH-LEAN jump - period duration measurement
Bit1 FAT-LEAN jump - period measurement
Bit2 B_LA_ALTER_US1_FETT - Probe voltage NKAT1 is in bold before SA
Bit3 B_LA_ALTER_US2_FETT - Probe voltage NKAT2 is in bold before SA
Bit4 B_LA_ALTER_DIAG1 - general. Diagnosis/Bank1 is active
Bit5 B_LA_ALTER_DIAG2 - general. Diagnosis/Bank2 is active
Bit6 B_LA_ALTER_SA_OK1 - Probe sp- NKAT1 is rich, but not above the MAX rich threshold
Bit7 B_LA_ALTER_SA_OK2 - probe sp- NKAT2 is in Fat, but not above MAX fat threshold

Variables:

Name	Meaning of	type	resolution
la_alter_st	status variable for TV monitoring Diagnostic	uc -	
la_vkat1 / 2_tv_ed; la_vkat1 / 2_p_ed; la_vkat1 / 3_spr_ed;	variable for VKAT monitoring TV shift Periodendauerüberwachung Srung time monitoring, stroke monitoring	uc -	
tkatm	temperature of the catalyst	your °C	
la_alt_p_st	status variable for period duration monitoring	uc -	
la_alt_p_mess_st	additional status variable for period duration	uc -	
la_alt_p_mess_of	measurement period duration without filtering period	your ms	
la_alt_p_mess	duration with filtering averaged, integrated TV shift	your ms	
lank_i_tv_gem	NKAT1/2 number of interference suppress.	sw ms	
la_alt_p_anz_spr	d. period.continuous measurements to the	uc -	
la_alt_p_anz_diag	number d. Period. Duration measurements for diagnosis TV	uc -	
la_p_tv1 / 2	shift, which is effective, is calculated from the actual period duration	sw ms	

la_alt_spr_st	Status for jump time diagnosis	uc -	
usv_spr_time_fett[]	Jump time from FAT -> LEAN	uc ms	
usv_spr_time_lean[]	Jump time from LEAN -> FAT	uc ms	
usv_spr_max_ref[]	Reference threshold max = 90% of the signal swing	your mV	
usv_spr_min_ref[]	Reference threshold min = 10% of the signal swing	your mV	
la_alt_spr_anz_m/f[]	Number of jump time measurements	uc -	
la_alt_spr_m/f_limit[] theoretical probe time based on actual theoretical time		uc ms	
usv_spr_m/f_quot_sum totaled measure of quality usv_spr_mager/bold_ge		your --	
		the	
m	average quality measure	your --	
la_alt_nkat_st	Status for NKAT probe diagnosis on integrated	uc -	
la_alt_s_ml	ML by CAT at SA	your kg	
la_alter_s_count1 / 2	Error counter for NKAT diagnosis	uc -	
la_alter_s_tv1 / 2	additional TV shift for NKAT diagnosis on integrated ML by	uc ms	
la_alt_s_we_ml la_nkat1 /	CAT at WE	your kg	
2_s_ed	Diagnostic variable for NKAT monitoring	uc -	

Application data:

Name	Type	importance
K_LA_OBD_RELEASE AND	constant	this is used to release the individual diagnoses BIT0: TV surveillance BIT1: Periodendauerüberwachung BIT2: NKAT probe diagnosis BIT3: Trim Control Bank1 BIT4: Trimming Regulation Bank2 BIT5: --- BIT6: CAT conversion BIT7: Jump time monitoring
K_LA_ALT_TV_MAX_COUNT	constant	MAX threshold for diagnostic counter - error entry threshold
K_LA_ALT_TV_MIN_COUNT	constant	MIN threshold for diagnostic counter - error entry threshold
K_LA_ALT_TV_INC	constant	Increment for diagnostic counter TV monitoring
K_LA_ALT_TV_DEC	constant	Decrement for diagnostic counter TV monitoring
K_LA_ALT_TV_MAX Constant		MAX diag black for TV shift
K_LA_ALT_TV_MIN	constant	MIN diag black for TV shift
K_LA_ALT_P_TEMP	constant	exhaust gas temperature threshold
K_LA_ALT_P_TAU	constant	Filtering constant period duration monitoring
K_LA_ALT_P_VERZ_T Constant		Delay time " lower N-
K_LA_ALT_P_N_MIN Constant		threshold " upper N-
K_LA_ALT_P_N_MAX Constant		threshold " lower RF-
K_LA_ALT_P_RF_MIN Constant		threshold " upper RF -threshold
K_LA_ALT_P_RF_MAX	constant	"
K_LA_ALT_P_ANZ_SPR	constant	Number to suppress interference
K_LA_ALT_P_ANZ_DIAG	constant	Number for diagnostic duration
K_LA_ALT_P_MAX	constant	upper diag.black for period. duration monitoring
K_LA_ALT_P_MIN	constant	lower diag.black for period. duration monitoring
K_LA_ALT_P_MAX_KONV	constant	upper diag.black for period. duration monitoring - with CAT CONV ERROR
K_LA_ALT_P_MIN_KONV	constant	lower diag.black for period. duration monitoring at KAT-KONV_ERROR
KF_LA_ALT_P_FAK_N_RF	map	Weighting map for monitoring period.duration
K_LA_ALT_P_TE_SPU Constantly		min. flushing time for TE before diagnosis starts

K_LA_ALT_SPR_ANZ_ LARGE / FAT	constant	Number of jump time measurements in BOLD or THIN
K_LA_ALT_SPR_M / F_ QUOT	constant	quality threshold for the jump times error thresholds
K_LA_USV_SPR_P_S PR	constant	Number of P jumps before diagnosis becomes active
K_LA_ALT_SPR_HYS	Constant	Hysteresis to detect RICH/LEAN spikes
K_LA_ALT_SPR_TEM P	constant	TABG threshold for diagnosis release
K_LA_ALT_SPR_N_MI N	constant	Nmin threshold for diagnostic window
K_LA_ALT_SPR_N_M AX	constant	Nmax threshold for diagnostic window
K_LA_ALT_SPR_RF_ MIN	constant	RFmin threshold for diagnostic window
K_LA_ALT_SPR_RF_ MAX	constant	RFmax threshold for diagnostic window
KF_LA_ALT_SPR_MA GER_LIMIT	constant	theoretical jump time - LEAN
KF_LA_ALT_SPR_FET T_LIMIT	constant	theoretical jump time - BOLD
K_LA_ALTER_US_FE TT_MAX	constant	max. threshold for BOLD position d. NKAT probe signal
K_LA_ALTER_US_FE T	constant	max. threshold for probe position
K_LA_ALTER_S_NMIN	Constant	lower N-threshold for diag. window
K_LA_AGE_S_NMA X	constant	upper N threshold for diag. window
K_LA_ALTER_S_TML	constant	min. motor running time for diag. window
K_LANK_TKAT_ BLACK	constant	minimum CAT temperature for diag. window
K_LA_ALTER_S_SA_T	Konstante	Minimum duration for defined SA and examination
K_LA_ALTER_S_WE_ ML	constant	Air volume threshold for testing according to WE
K_LA_ALTER_S_ML	constant	Air volume threshold for testing in overrun
K_LA_ALTER_S_SA_ US	constant	Probe voltage threshold for testing in overrun or WE
KL_LA_ALTER_S_TV	characteristic	additional enrichment for WE test, dep. from ml
K_LS_ALTER_S_WE_ T	constant	Waiting time without TV shift for WE check
K_LS_ALTER_S_TV_T	constant	Waiting time with TV postponement for WE check
K_LA_ALTER_S_COU E.G	constant	Threshold for error counter until error entry NKAT probe
K_LA_ALT_S_TKATM	Constant	CAT temp threshold for vibration testing