SRT / SRAE SOFTWARE DOCUMENTATION SUPER-USER

10. TURBO

chapter 10

Release : 5

REVISIONS DOCUMENT

Release	*Author*	*Date*	*Modifications*
1.0	M.Mersier	04/09/2004	• Creation (SRA)
2.0	M.Mersier	02/09/2006	• Add Close Loop (SRT/SRAE)
2.1	M.Mersier	03/17/2006	• Add Diagnostic Nturbo • Add Test functions for wastegate and contra-pressure output electrovalves (SRT/SRAE)
2.2	M.Mersier	09/08/2006	• TransferTurbo Enable and N_turbo selection and configuration in Client_Configuration.doc • Transfer N_turbo input Acquisition in Input.doc)
4.1	E.Girardeau	13/11/2009	• The Turbo strategies are disabled (On and after Software Release 4.16).
5	E.Girardeau	07/12/2009	• Delete water injection management • Turbo strategies improvement

10.1 Turbo Environment Definition

10.1.1 Waste-gate 1A Control

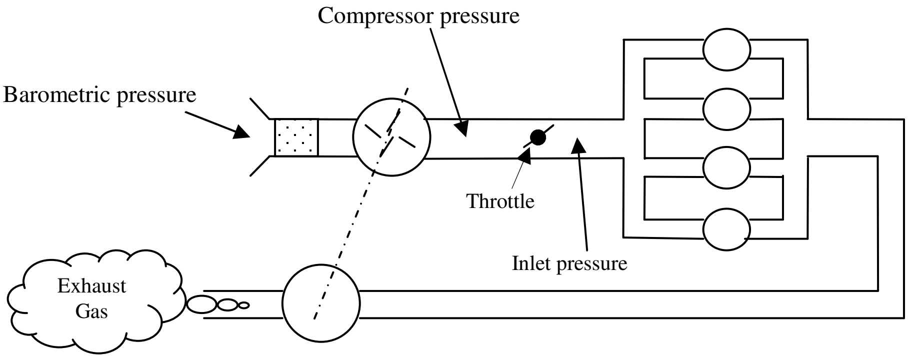
If the turbo function is selected, by **EE.CfgU.Turbo_EN** (see Strategies authorisation in documentation Client_Configuration.doc), and if the output for the waste-gate valve is selected by **EE.CfgU.OutputWasteGate1A** (see Outputs selection and assignment in documentation Client_Configuration.doc) ,it is necessary to configure the drive frequency of the wastegate valve in the map **EE.Tbo.FreqWasteGate1A** (Unit : Hz)

10.1.2 Waste gate 1B (or Contra Pressure) Control (optional)

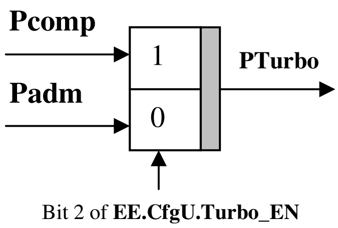
If the turbo function is selected, by **EE.CfgU.Turbo_EN**, and if the output for the waste gate 1B valve is selected by **EE.CfgU.OutputWasteGate1B** (see Outputs selection and assignment in documentation Client_Configuration_x.doc) ,it is necessary to configure the drive frequency of the valve in the map **EE.Tbo.FreqWasteGate1B** (Unit : Hz)

10.1.3 Pressure inputs

The turbo strategy can manage two kinds of pressures: the compressor pressure (Pcomp) and the inlet pressure (Padm). The engine requirements (Injection times and Ignition advance) are always computed in accordance with the inlet pressure. But the turbo control can be function of compressor pressure or inlet pressure.



Thus the user chooses which pressure input to use. That selection is done in the **EE.CfgU.Turbo_EN** map and specifically depends on the value of the 3rd bit. (See Strategies authorisation in documentation Client_Configuration.doc)



The selected pressure value is then copied in the **PTurbo** CLL channel.

10.1.3.1 Boost Pressure breakpoints (Inlet pressure)

The maps for the engine requirement (Injection Time, Ignition advance) use breakpoints of boost pressure (turbo-charged inlet air pressure). This pressure is labelled as P2.

The table **EE.Bkps.P2**can have up to 24 breakpoints, expressed in Bar, and is used for addressing turbo maps as a function of boost pressure.

The number of breakpoints may be changed by the map table **EE.CfgU.NbkpP2**

10.1.4 Perfo Switch

A dashboard mounted switch allows the selection of two sets of turbo control maps, if fitted. The two positions are named High and Low.
(See § Inputs selection and assignment in documentation Client_Configuration.doc for Input selection)

The following maps are dependant on the perfo Switch:

Pressure regulation target	**EE.TboBf.tab_TgtPTurbo**
Turbo speed regulation target	**EE.TboBf.tab_TgtNTurbo**
Open loop turbo command	**EE.Tbo.tab_CmdTurbo_BO**

10.2 Turbo Working Mode

1. **Normal Working Mode**
   Standard operation mode from which the engine requirements (ignition advance, injection quantities) and turbo commands result from the basic turbo setting.

   Two kinds of turbo regulation are available:
   - ✓ Open Loop Only
   - ✓ Open Loop plus Close Loop

2. **Upshift**
   Apply a specific turbo command during upshift sequence

3. **Start BangBang**
   Use several specific commands (ignition advance, injection quantities, turbo command) in the aim to maintain a sufficient turbo speed to guarantee a strong start.

4. **LiftOff BangBang**
   Apply several specific commands (ignition advance, injection quantities and turbo waste gates commands) to maintain a sufficient turbo speed when the driver lifts off.

5. **Test**
   Apply predefined waste gate RCO for actuators' testing.

6. **Default**
   Failsafe waste gates' commands.

The **St_WG** CLL channel content corresponds to the current turbo working mode. The following values are available:

Channel Value	Enumeration	Working mode
0	OFF	Turbo inactive
1	BangStart	Start Bangbang
2	BangLiftOff	LiftOff Bangbang
4	BO	Open Loop Command
8	Halt	Engine stall
16	Default	FailSafe Mode
32	Up	Upshift
64	BF_P	Pressure regulation
128	BF_N	Turbo speed regulation

10.2.1 Turbo Status

The **Turbo** CLL channel summarizes the global status of the turbo charger management. It is a bitmapped channel whom each bit signification is detailed below:

- ✓ **Bit 0:** Global Turbo Status

Bit Value	0	1
Meaning	Inactive	Active

- ✓ **Bit 1:** Turbo Performance

Bit Value	0	1
Meaning	High performances	Low performance

- ✓ **Bit 2:**
  Not Used.

- ✓ **Bit 3:** Pulverisation status

Bit Value	0	1
Meaning	OFF	ON

- ✓ **Bit 4:** Pressure Regulation

Bit Value	0	1
Meaning	Inactive	Active

- ✓ **Bit 5:** Turbo Speed Regulation

Bit Value	0	1
Meaning	Inactive	Active

- ✓ **Bit 6:** Open loop command

Bit Value	0	1
Meaning	Inactive	Active

- ✓ **Bit 7:** FailSafe mode

Bit Value	0	1
Meaning	Inactive	Active

10.2.2 Start Bangbang (5 mS)

This option is selected via the map **EE.CfgU.Turbo_EN** (See § Strategies authorisation in documentation Client_Configuration.doc), and is enabled only when the Turbo performance is high.

The Start Bangbang function corresponds to a phase of post combustion (obtained by the application of an ignition retard applied to a specific quantity of fuel and a specific open loop turbo command), with the intention of maintaining a turbo speed sufficient to guarantee charging the engine.

10.2.2.1 Start Bangbang Activation Switch

A dashboard-mounted switch allows the activation of the start bangbang function (only if the turbo performance state is high).
(See § Inputs selection and assignment in documentation Client_Configuration.doc for Input selection EE.CfgU.SwBangDepInput)

If the switch is not fitted, the Start Bangbang function is not applied and the error code 53 is set in ErrConfig.
The activation of this switch will not activate this strategy if the vehicle speed is greater than the threshold **EE.Lim.SpeedMaxStartLim** , to avoid taking into account unintentional presses of the steering wheel switch while driving.

10.2.2.2 Strategy Deactivation

The Start Bangbang function is deactivated by releasing the switch (stopping the post combustion and the start limiter) or if the Perfo switch state is set to the Low. On the other hand, if the switch is held active, and the engine speed exits the activation band or the vehicle speed is greater than the threshold **EE.Tbo.SVitOutBangDep**, then the post combustion is stopped but the start limiter is maintained until the switch is released.

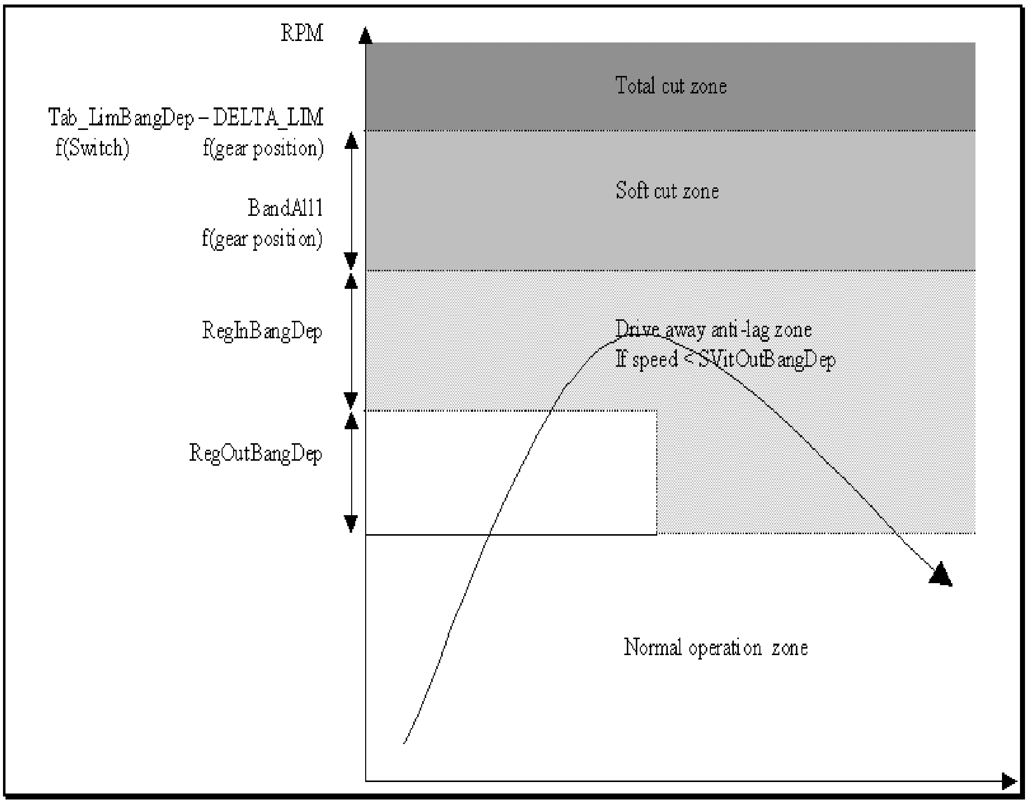
10.2.2.3 Start Bangbang Limiter and Anti-Lag

The start phase is coupled with different levels of engine speed limiter selected by a dashboard-mounted 5-position switch. That dashboard-mounted switch allows the selection of five engine limiter levels for Start control procedure (See § Inputs selection and assignment in documentation Client_Configuration.doc for Input selection **EE.CfgU.RotacLimDep_IN**)
In case of that five positions switch is not fitted, the Start Bangbang function is disabled.
The table **EE.Tbo.RotacLimDep** has five pairs of voltage values (minimum and maximum) defining the electrical limits of each switch position.

Each of the five switch positions is associated with an engine speed limiter defined by the map **EE.Tbo.tab_LimBangDep.** These engine speed limiters replace the absolute limiters (described in documentation Speed_Limiters.doc) and all the connected functions apply here as well (DELTA_LIM f(gear); soft cut bands f(gear), …).

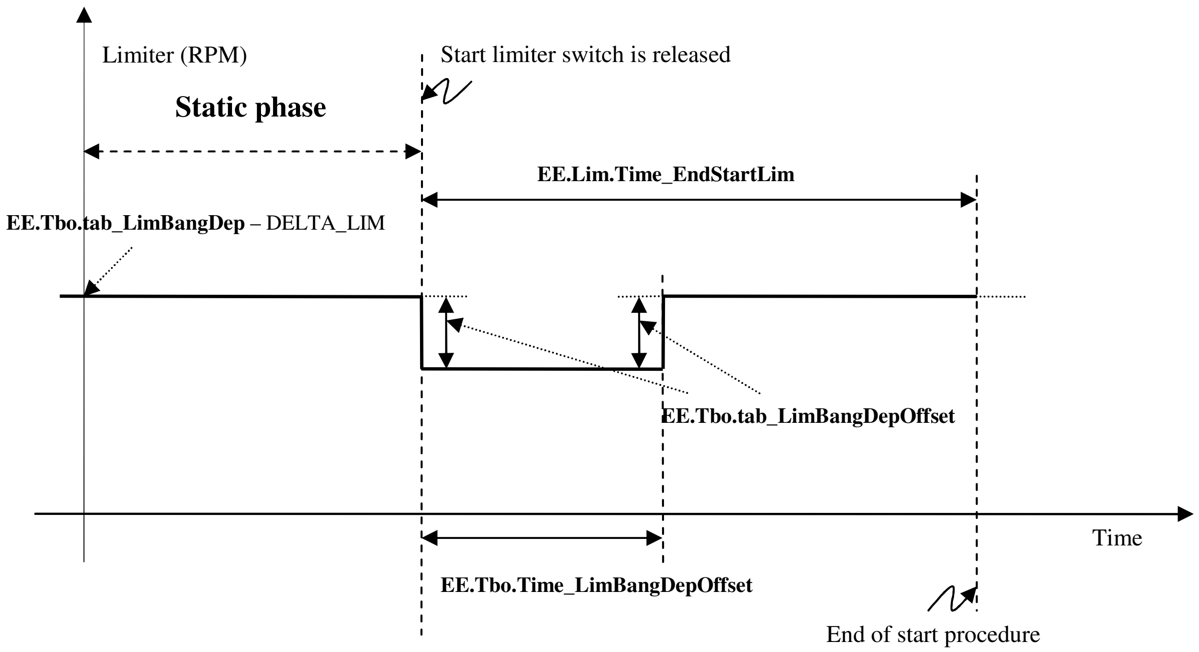
The start Bangbang is not applied if the engine speed is lower than the threshold **EE.Tbo.RegInBangDep** expressed in RPM below the “start limiter” :

The band of RPM **EE.Tbo.RegOutBangDep** below start bangbang entry RPM for exiting start bangbang



10.2.2.3.1 Extended Engine Speed Limitation

The limiters applied during the static start procedure can be time extended after start switch releasing. The user defines the total delay via the map **EE.Lim.Time_EndStartLim** which is indexed following the position of the end time start limiter five positions switch. (See § Inputs selection and assignment in documentation Client_Configuration.doc for Input selection **EE.CfgU.RotacTimeEndLimDep_IN**)
Moreover, it is possible to define a limiter offset **EE.Tbo.tab_LimBangDepOffset** (positive or negative value) that is applied just after the static phase end. The offset’s application duration is user configurable via the user map **EE.Tbo.Time_LimBangDepOffset .**



Note: The global start control does not need two distinct 5 positions switches (RotacLimDep and RotacTimeEndLimDep). It is enough to select the same ECU input and to configure them in the same way

10.2.2.4 Bangbang Pressure Breakpoints

The Bangbang mode maps are addressed by 5 specific breakpoints of boost pressure in Bar, defined in the table **EE.Bkps.P2bang.**

10.2.2.5 Bangbang Engine Speed Breakpoints

The Bangbang mode maps are addressed by 8 breakpoints of engine speed in RPM, defined in the table: **EE.Bkps.RegBang**

10.2.3 Lift-off Bangbang

This option is selected by the map **EE.CfgU.Turbo_EN** (See § Strategies authorisation in documentation Client_Configuration.doc), and is activated only if the Start Bangbang function is not active and if the turbo performance status is in High.

10.2.3.1 Application Conditions

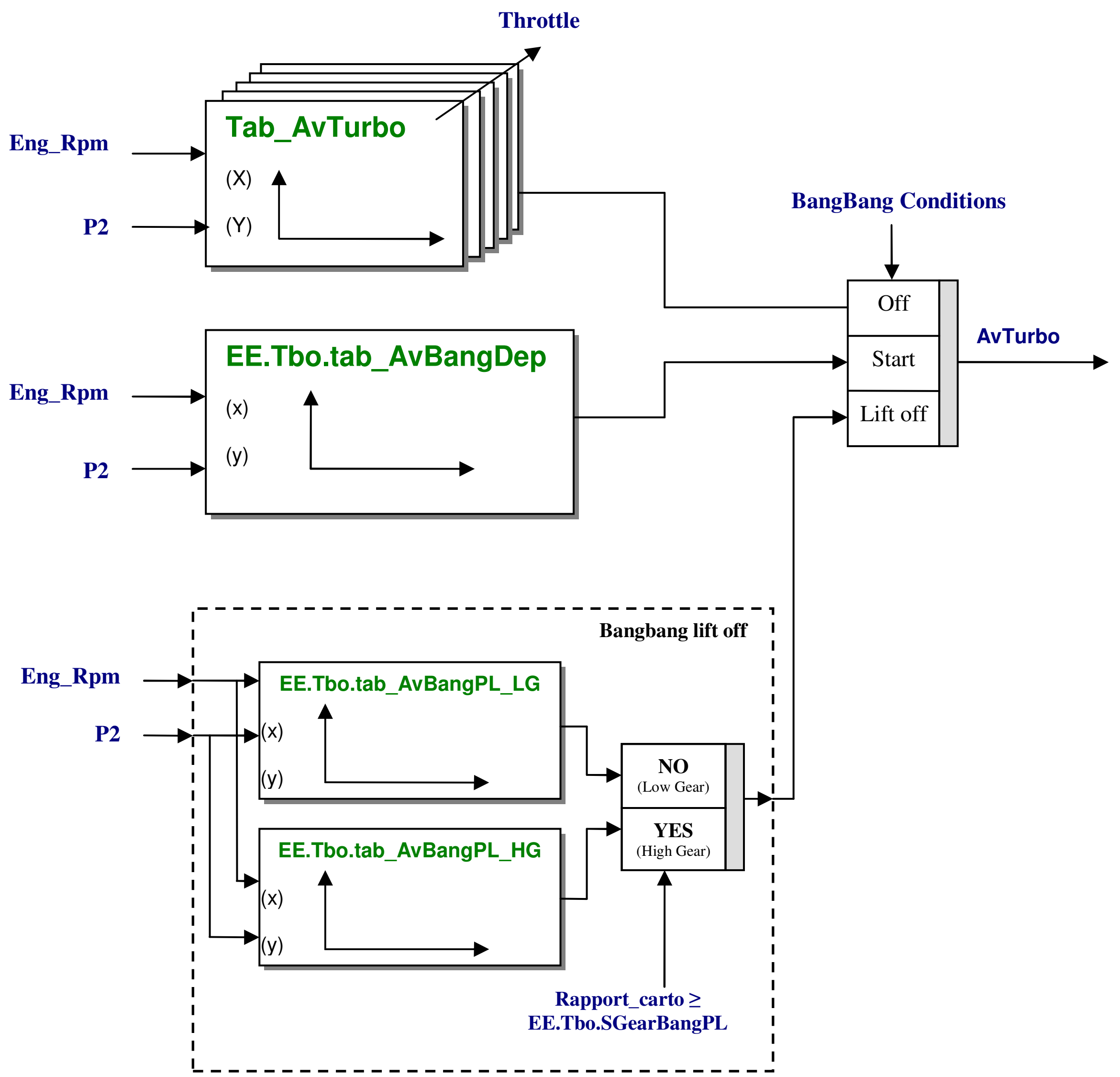
To be active, the engine speed must be greater than the threshold **EE.Tbo.SRegBangPL** and the throttle position must be lower than the threshold **EE.Tbo.SPapBangPL** .

The lift off Bangbang function is exited if the engine speed becomes lower than the above threshold minus the hysteresis **EE.Tbo.HystPapBangPL**, or if the throttle position becomes higher than the function entry threshold plus the hysteresis **EE.Tbo.HystRegBangPL**, or if the Turbo performance status switches to Low performance.

10.3 Engine requirements

10.3.1 IGNITION ADVANCE (5 mS)

Ignition advance computation Graph:



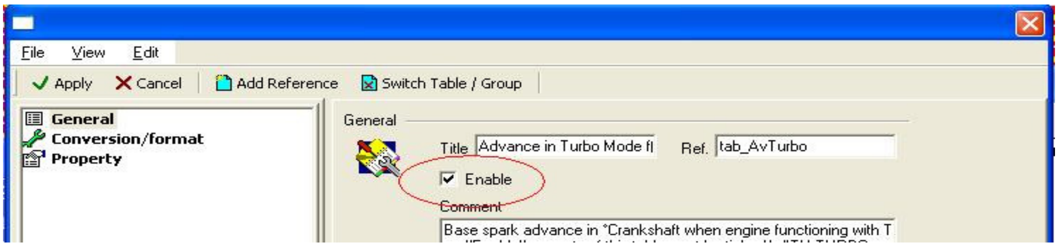
10.3.1.1 Normal Ignition Advance

In normal working mode (out of bangbang) the base ignition advance is defined by the 3D map **tab_AvTurbo** as a function of engine speed and boost pressure for X and Y axis where breakpoints numbers are specified in the maps **EE.CfgU.NbkpRegShort** and **EE.CfgU.NbkpP2**. The 3rd axis (Z axis) is throttle position indexed with 5 breakpoints which are defined in the map **EE.Bkps.ThrottleShort**.

This map replaces the ignition advance map described in the documentation Ignition.doc. The advance corrections are the same than the ones that adjust the standard advance also defined in the Ignition.doc file.

WARNING : When using this map , it's necessary to enter the map properties to ***ENABLE*** the map. Moreover, the both ignition advance maps **EE.Ign.tab_avance** and **EE.Ign.tab_avance1** included in the map file IGNITION group must be ***DISABLED***

Just a quick reminder on the way to toggle the "enable" tag in the map properties:



.

10.3.1.2 Bangbang

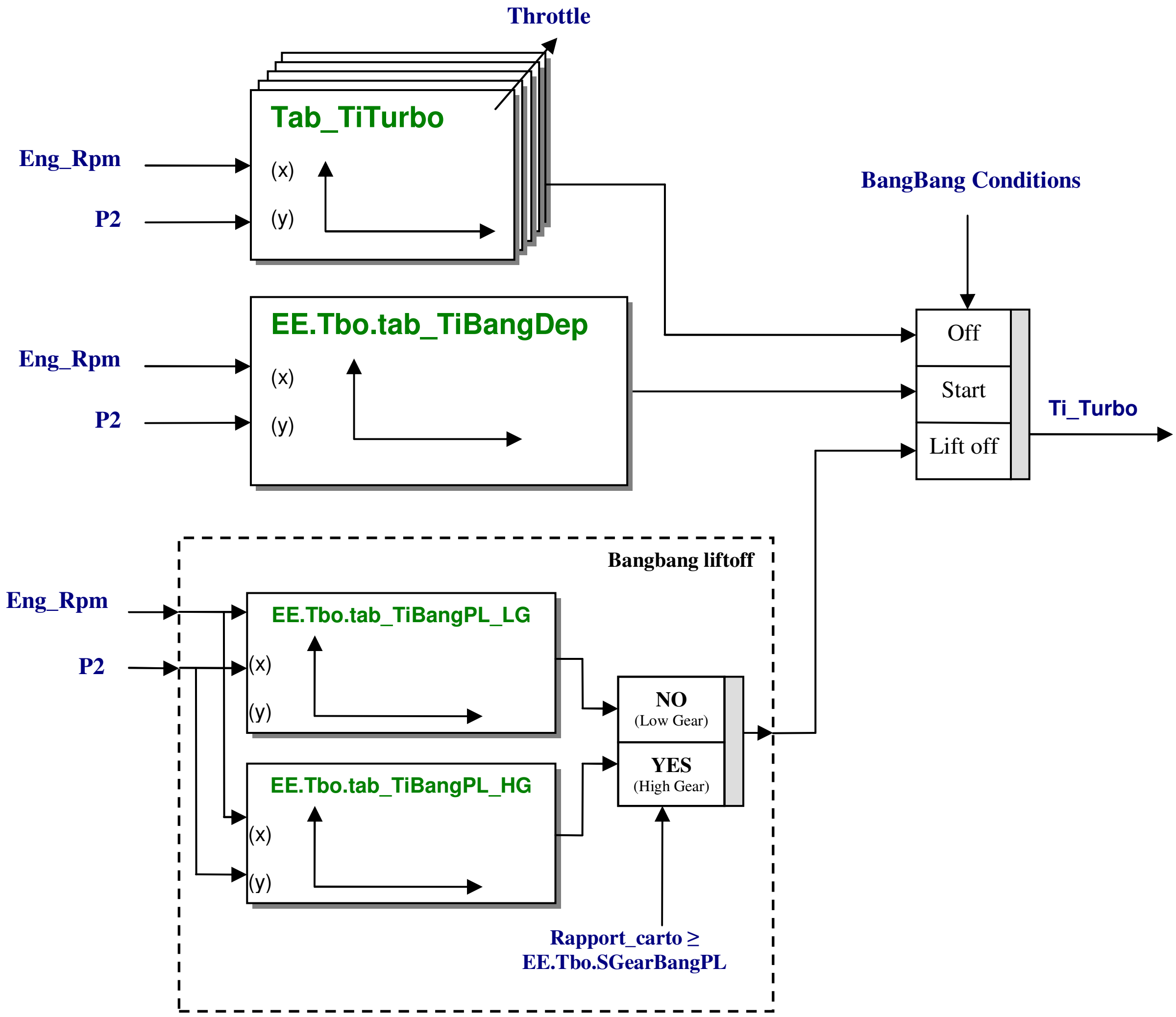
When the bangbang conditions are met, a specific set of maps is handled to compute the ignition advance. So, according the bangbang status (**St_WG** = BangStart or BangLiftOff) one of the following maps is used: **EE.Tbo.tab_AvBangPL_HG** or **EE.Tbo.tab_AvBangPL_LG** or **EE.Tbo.tab_AvBangDep**.

Antilag state	St_WG value	Rapport_carto value	Used ignition map
Start Bangbang	BangStart (value: 0x01)		**EE.Tbo.tab_AvBangDep**
LiftOff Bangbang	BangLiftOff (value: 0x02)	Greater or equal to **EE.Tbo.SGearBangPL**	**EE.Tbo.tab_AvBangPL_HG**
		Lower than **EE.Tbo.SGearBangPL**	**EE.Tbo.tab_AvBangPL_LG**

In any case, the ignition advance is function of engine speed and boost pressure (Inlet pressure) for X and Y axis where breakpoints number are specified in the maps **EE.CfgU.NbkpRegShort** and **EE.CfgU.NbkpP2**

10.3.2 INJECTION QUANTITY (5 mS)

Fuel quantities computation Graph:

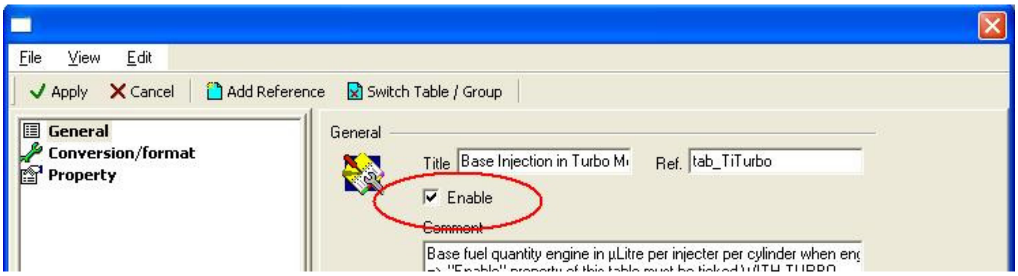


10.3.2.1 Normal Injection Quantity

In normal working mode (out of bangbang), the base injection quantity is defined by the 3D map **tab_TiTurbo** as a function of engine speed and boost pressure for X and Y axis where breakpoints number are specified in the maps **EE.CfgU.NbkpRegShort** and **EE.CfgU.NbkpP2**. The 3rd axix (Z Axis) is throttle position indexed with 5 Breakpoints which are defined via the map **EE.Bkps.ThrottleShort**.

That advance map replaces the two injection maps described in the documentation INJECTION.doc. The applied corrections are the one used to adjust the normal injection time also defined in the INJECTION.doc file.

WARNING : When using this map , it is necessary to ***ENABLE*** the map by modifying its properties



The both other injection maps (**EE.Ti.TabTinj** and **EE.Ti.TabTinj1**) in the INJECTION group must be ***DISABLED***

10.3.2.2 Bangbang

When the bangbang conditions are met, a specific set of maps is handled to compute the fuel quantities. So, according the bangbang status (**St_WG** = BangStart or BangLiftOff) one of the following maps is used: **EE.Tbo.tab_TiBangPL_HG** or **EE.Tbo.tab_TiBangPL_LG** or **EE.Tbo.tab_TiBangDep**.

Antilag state	St_WG value	Rapport_carto value	Used ignition map
Start Bangbang	BangStart (value: 0x01)		**EE.Tbo.tab_TiBangDep**
LiftOff Bangbang	BangLiftOff (value: 0x02)	Greater or equal to **EE.Tbo.SGearBangPL**	**EE.Tbo.tab_TiBangPL_HG**
		Lower than **EE.Tbo.SGearBangPL**	**EE.Tbo.tab_TiBangPL_LG**

In any case that specific injection quantity is function of engine speed and boost pressure (Inlet pressure) for X and Y axis where breakpoints numbers are specified in the maps **EE.CfgU.NbkpRegShort** and **EE.CfgU.NbkpP2**.

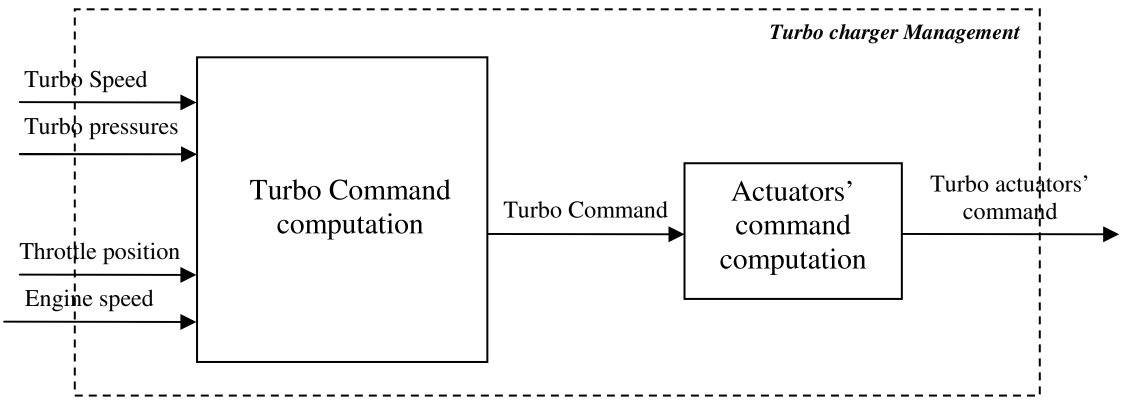
10.4 Turbo Charger Management

The ECU (SRT or SRAE) can handle only one turbo charger body with its accessories (FreshAir, and intercooler spray). The turbo charger body is driven via the both actuators named waste gate 1A and waste gate 1B. The both turbo actuators' commands (**RCO_WG1A** & **RCO_WG1B**) square with a PWM duty cycle whom range value is clipped [0%, 100%].

<u>Nota:</u> Even if it is not managed in the same way, the wastegate 1B corresponds to the contra pressure actuator in the old turbo management.

The global turbo management can be split in two functions. The first one (the main one) is aimed to compute the global turbo command and the second computes the both actuators' commands as a function of the turbo command.

The global computation path can be schematized as follow:



The turbo command corresponds to an amount of global command that must be applied to the regulation turbo actuators (the both waste gates). Thus that amount of command which is ranged from 0% to 100% is at last treated through a set of conversion maps to achieve the final waste gate RCO command.

10.4.1 Turbo Command Computation (5 mS)

10.4.1.1 Turbo mode

The following turbo working modes are available:

1. <u>Normal mode</u>
   Standard working mode

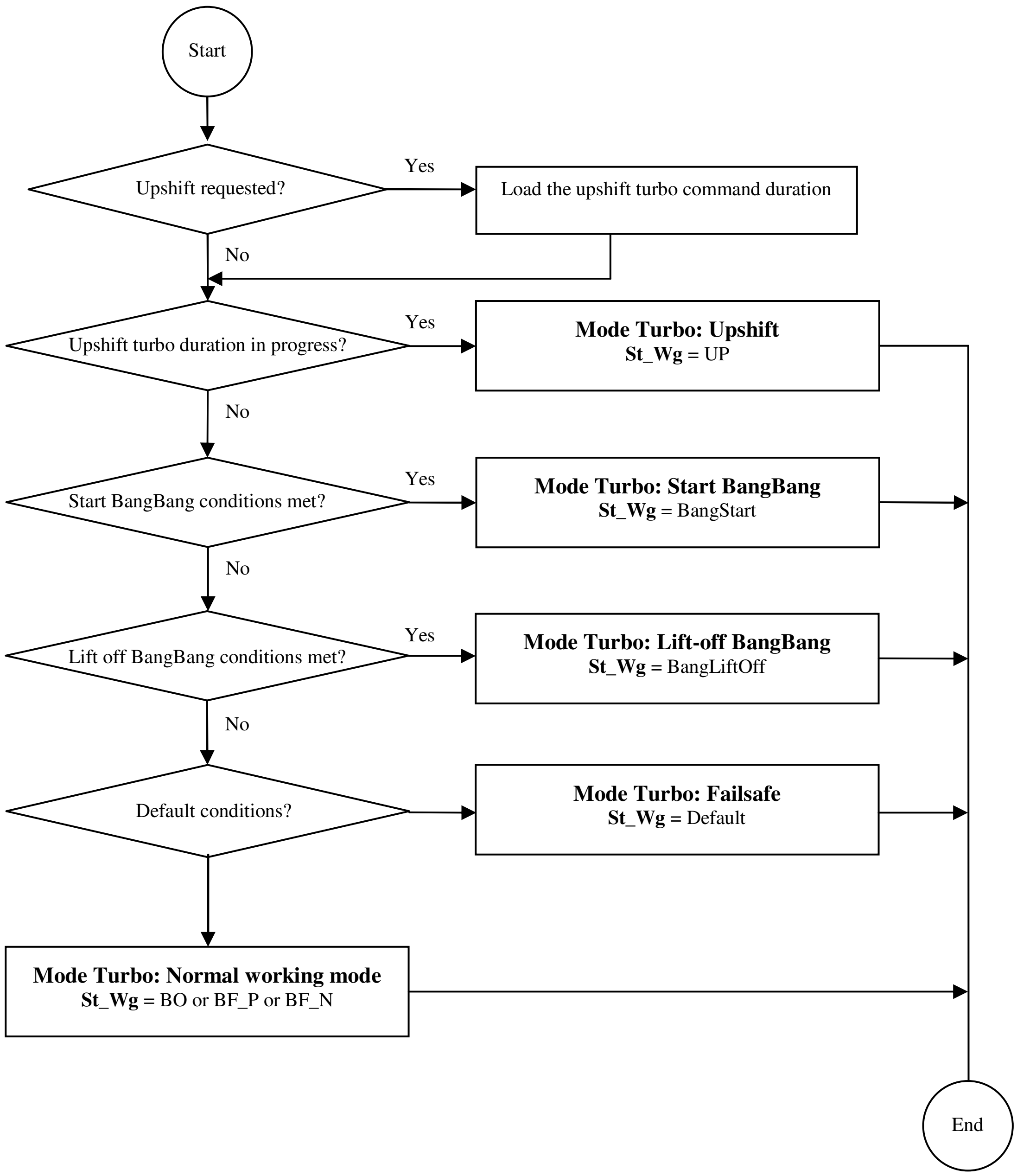
2. <u>Upshift</u>
   Defines a specific turbo command applied during upshift command

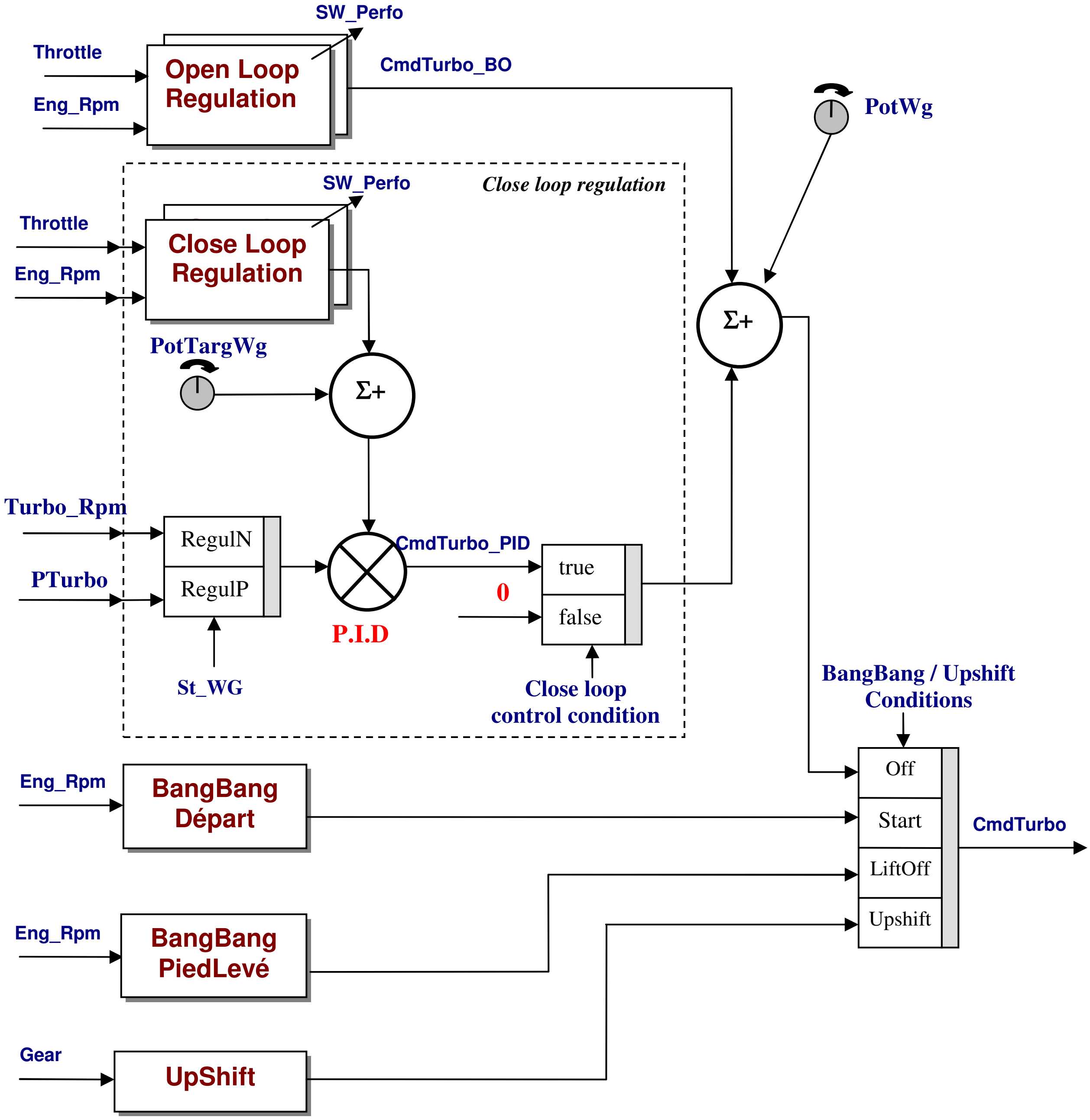
3. Start Bangbang
   Reduce the turbo lag during race start processing

4. Lift off bangbang
   Reduce the turbo lag when the driver lifts off his foot from the gas pedal.

5. FailSafe
   Safety mode

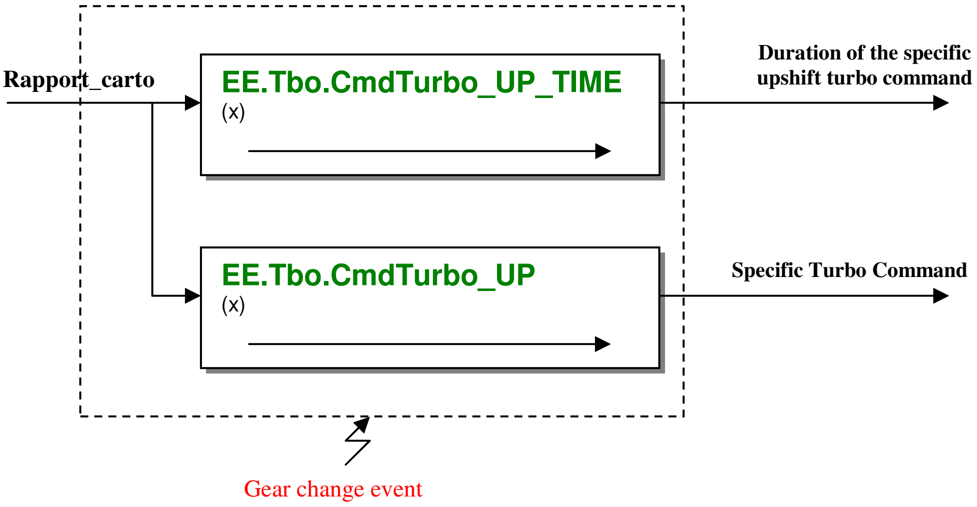
Turbo mode determination:



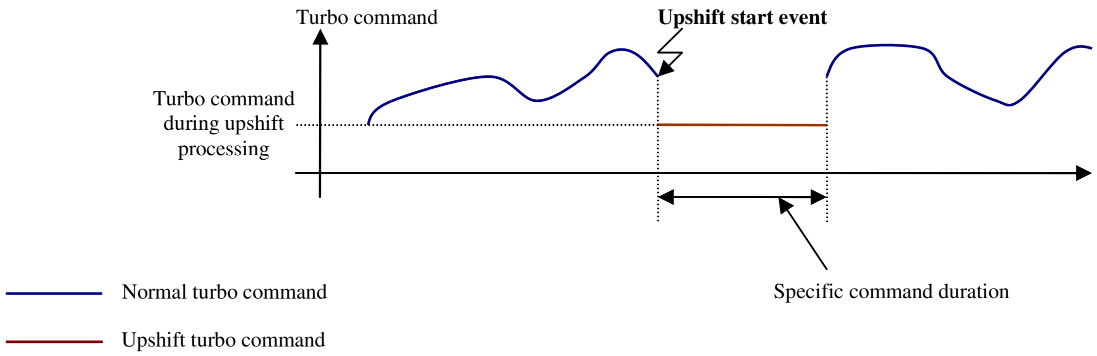


10.4.1.2 UpShift command

On gear change, the user can apply a specific turbo command whom value and timing depend on the initial gear position when the shift gear is requested.



There is no dedicated CLL channels linked to the above interpolations processing. Thus when the upshift logic starts, the Turbo command (**CmdTurbo** CLL channel) switches to the specific computed command during the computed duration.



10.4.1.3 Start Bangbang Waste Gate Regulation

In Start bangbang mode, the waste gate control is issued from the table **E.Tbo.tab_RcoWGbangDep**, addressed as a function of engine speed:

10.4.1.4 Lift Off Bangbang Turbo Command

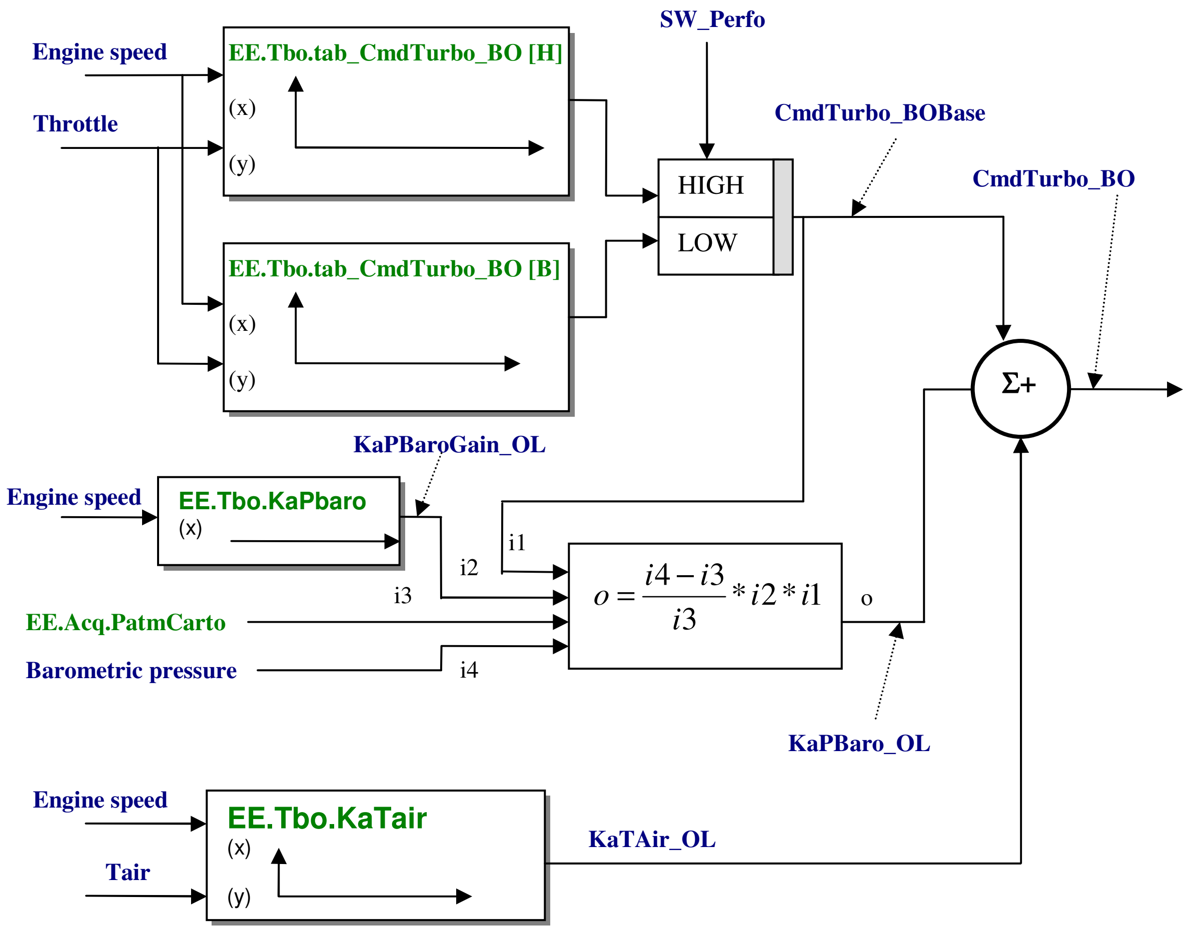
In Lift Off Bangbang mode, the turbo command is issued from the table **EE.Tbo.tab_RcoWGbangPL**, addressed as a function of engine speed .

10.4.1.5 Normal Turbo Command

In case of normal working mode (Neither bangbang nor upshift), the turbo command is the computation result of the Open Loop command plus a Close Loop command when close loop mode is enabled.

10.4.1.5.1 Open Loop Command

Whatever the turbo normal mode (close loop or open loop), the open loop is always computed. In case of close loop, the final turbo command corresponds to the open loop command plus the close loop command. That command is a direct amount of turbo command which is expressed in % and is ranged from 0% to 100%.
The computation path matches with that following diagram:



Thus, according the performance switch status, the basic open loop command **CmdTurbo_BOBase** is issued from one of the both maps **EE.Tbo.tab_CmdTurbo_BO[H]** or **EE.Tbo.tab_CmdTurbo_BO[L]**

Performance switch status	Used Open loop map
High	**EE.Tbo.tab_CmdTurbo_BO[H]**
Low	**EE.Tbo.tab_CmdTurbo_BO[L]**

In any cases the basic open loop command is a function of engine speed and throttle position for X and Y axis where breakpoints tables are **EE.Bkps.RegShort** and **EE.Bkps.ThrottleMiddle**

Nota: The PTA file includes only one open loop command user map (**EE.Tbo.tab_CmdTurbo_BO**) but it is physically a 3 dimensional mal whom third axis corresponds to the performance status.

Two corrections are applied to that basic open loop command.

- ✓ The first one (**KaPBaro_OL** cll channel) is aimed to correct the basic command, in accordance with the difference between the current barometric pressure and the referenced pressure (**EE.Acq.PatmCarto**). It is computed as follow.

$$KaPBaro\_OL = \frac{Barometric\_pressure - EE.Acq.PatmCarto}{EE.Acq.PatmCarto} * KaPBaroGain\_OL * CmdTurbo\_BOBase$$

The gain **KaPBaroGain_OL** weights the pressure ratio. It is defined by the 1D map **EE.Tbo.KaPbaro** as a function of the engine speed where breakpoints table is **EE.Bkps.RegShort_Turbo**.

Nota: The correction is computed only if the barometric pressure sensor is fitted and without sensing fault.

- ✓ The second correction (**KaTAir_OL**) is defined by the 2D map **EE.Tbo.KaTair** as a function of the engine speed and the air temperature for X and Y axis where breakpoints tables are **EE.Bkps.RegShort_Turbo** and **EE.Bkps.AirT_Turbo**.

Nota: The correction is computed only if the air temperature sensor is fitted and without sensing fault.

The final open loop turbo command (**CmdTurbo_BO**) is equal to the summation of the both corrections and the basic command.

$$CmdTurbo\_BO = CmdTurbo\_BOBase + KaPBaro\_OL + KaTAir\_OL$$

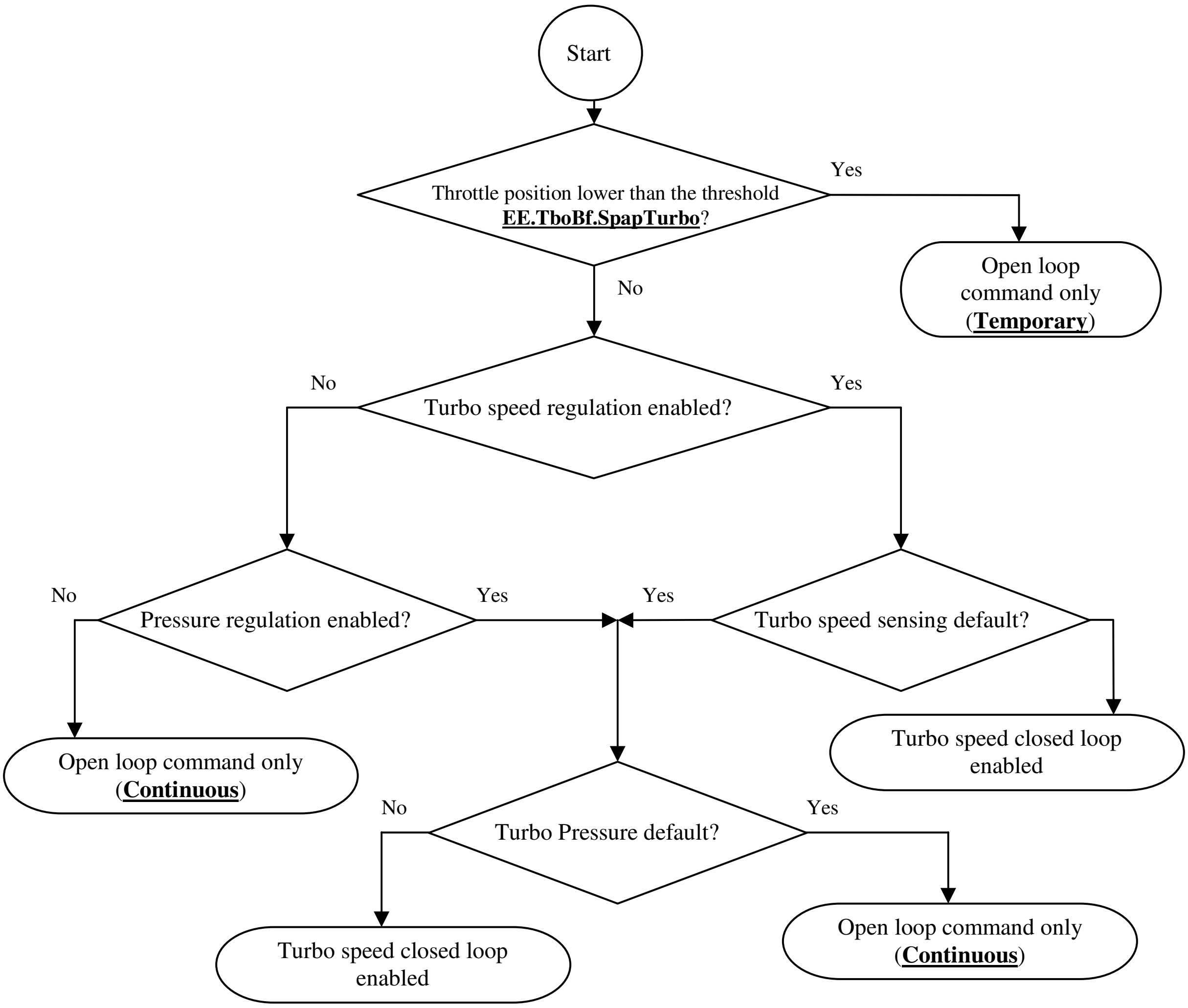
That final value is clipped to [0%, 100%].

10.4.1.5.2 Close Loop/Open loop Regulation

The system can handle two kinds of regulation: Turbo speed or Turbo pressure. Following the Turbo configuration map **EE.CfgU.Turbo_EN** as described in the document "Client_Configuration.doc", the user may choose only one regulation type or the both.

10.4.1.5.2.1 Close Loop Condition

The diagram below describes the regulation mode determination.



When the both regulation types are enabled, the main one is the turbo speed regulation and the second (pressure regulation) type is a kind of safety regulation mode. At last, if the both regulations types are disabled, the system switches *permanently* to the open loop mode.

The user may define a throttle position threshold that enabled/disabled the close loop. Thus when the throttle position is lower than that threshold, the regulation switches to the open loop mode. And as soon as the threshold is overstepped, the system swaps to the previous active close loop mode.

10.4.1.5.3 Pressure target

10.4.1.5.3.1 Turbo pressure target

The basic turbo pressure target **Tgt_PTurbo_Base** depends on the Switch Perfo status and is defined by the following 3D map **EE.TboBf.tab_TgtPTurbo** as a function of engine speed and throttle position, where breakpoints tables are **EE.Bkps.RegShort** and **EE.Bkps.ThrottleMiddle**
The third axis matches with the performance switch status.

That basic command is afterward adjusted with two additive corrections:

- ✓ The first one (**KaPBaro_CL_P**) is aimed to correct the turbo pressure target in accordance in accordance with the pressure difference between the current barometric pressure and the referenced pressure (**EE.Acq.PatmCarto**). It is computed as follow.

$$KaPBaro\_CL\_P = \frac{Barometric\_pressure - EE.Acq.PatmCarto}{EE.Acq.PatmCarto} * KaPBaroGain\_CL\_P * \text{Tgt\_PTurbo\_Base}$$

The gain **KaPBaroGain_CL_P** weights the pressure ratio. It is defined by the map **EE.TboBf.KaPbaro_PTurbo** as a function of engine speed where breakpoints table is **EE.Bkps.RegShort_Turbo**.

Nota: The correction is computed only if the barometric pressure sensor is fitted and without sensing fault.

- ✓ The second correction which is function of the air temperature (**KaTAir_CL_P**) is defined by the map **EE.TboBf.KaTair_PTurbo** as a function of engine speed and the air temperature.

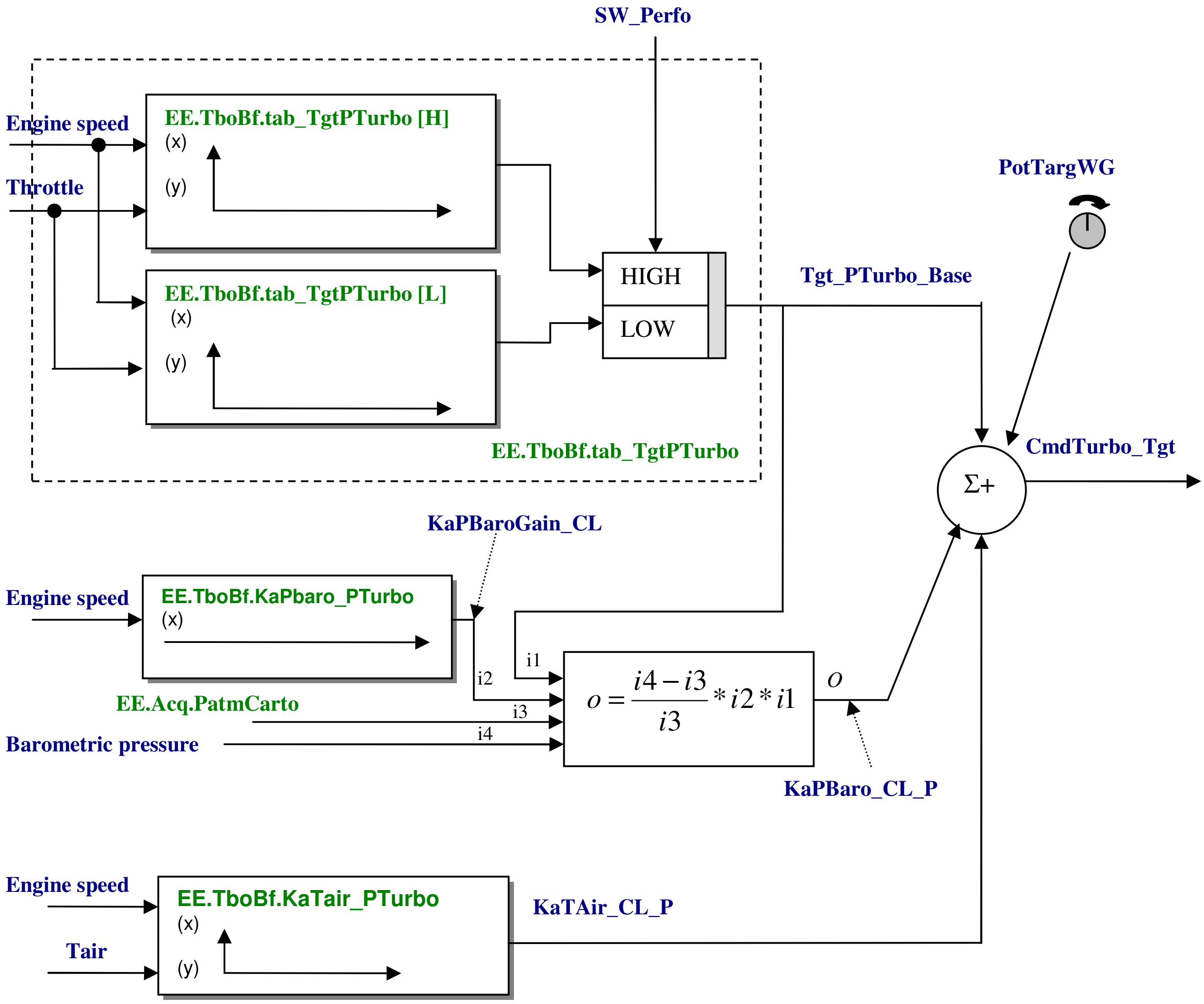
Nota: The correction is computed only if the air temperature sensor is fitted and without sensing fault.

The final pressure target is equal to the summation of the basic target (**Tgt_PTurbo_Base**) and the both target corrections (**KaPBaro_CL_P**, **KaTAir_CL_P**) and the vision potentiometer **PotTargWG**.

$$CmdTurbo\_Tgt = Tgt\_PTurbo\_Base + KaPBaro\_CL\_P + KaTAir\_CL\_P + PotT\arg WG$$

Nota: The CLL channel **CmdTurbo_Tgt** is the same one that is used in case of pressure regulation or turbo speed regulation. So, following the turbo regulation mode (CL or OL), the channel unit is mbar or Rpm.

The following diagram describes the target computation path.



Nota: the both drawn maps **EE.TboBf.tab_TgtPTurbo [H]** and **EE.TboBf.tab_TgtPTurbo [L]** are wrapped in a single map (**EE.TboBf.tab_TgtPTurbo**). The maps' suffixes [H] or [L] correspond to the third dimensional axis.

10.4.1.5.3.2 Turbo pressure error

The regulation error is equal to the difference between the pressure target computed above and the sensed pressure. That pressure reference is defined via the Turbo configuration map **EE.CfgU.Turbo_EN** which is described in the "Client_Configuration.doc" documentation file.

The error is computed as follow:

$$\text{CmdTurbo\_PID\_Err} = \text{CmdTurbo\_Tgt - PTurbo}$$

This value is used as input of the PID function.

Nota: The CLL channel **CmdTurbo_PID_Err** is the same one that is used in case of pressure regulation or turbo speed regulation. So, in accordance with the turbo regulation mode, the channel's unit is mbar or Rpm.

10.4.1.5.4 Turbo speed target

10.4.1.5.4.1 Turbo speed target

The basic turbo speed target **Tgt_NTurbo_Base** depends on the Switch Perfo status and is defined by the following 3D map **EE.TboBf.tab_TgtNTurbo** as a function of engine speed and throttle position, where breakpoints tables are **EE.Bkps.RegShort** and **EE.Bkps.ThrottleMiddle**
The third axis matches with the performance switch status.

That basic command is afterward adjusted with two additive corrections:

- ✓ The first one (**KaPBaro_CL_N**) is aimed to correct the turbo speed target in accordance with the pressure difference between the current barometric pressure and the referenced pressure (**EE.Acq.PatmCarto**). It is computed as follow:

$$KaPBaro\_CL\_N = \frac{Barometric\_pressure - EE.Acq.PatmCarto}{EE.Acq.PatmCarto} * KaPBaroGain\_CL\_N * \text{Tgt\_NTurbo\_Base}$$

  The gain **KaPBaroGain_CL_N** weights the pressure ratio. It is defined by the map **EE.TboBf.KaPbaro_NTurbo** as a function of engine speed where breakpoints table is **EE.Bkps.RegShort_Turbo**.

Nota: The correction is computed only if the barometric pressure sensor is fitted and without sensing fault.

- ✓ The second correction which is function of the air temperature (**KaTAir_CL_P**) is defined by the map **EE.TboBf.KaTair_PTurbo** as a function of engine speed and the air temperature.

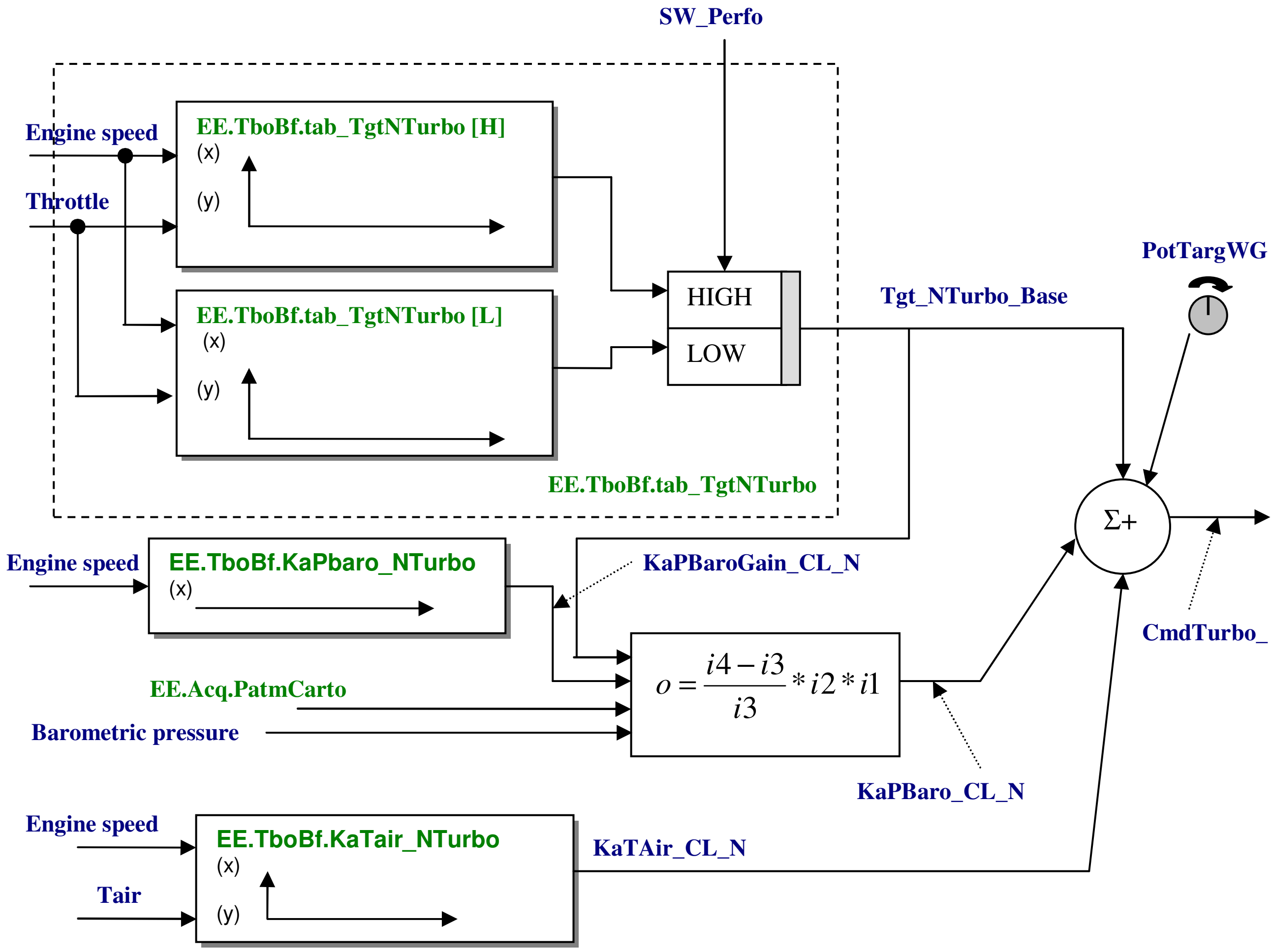
Nota: The correction is computed only if the air temperature sensor is fitted and without sensing fault.

The final turbo speed target is equal to the summation of the basic target (**Tgt_NTurbo_Base**) and the both corrections (**KaPBaro_CL_N**, **KaTAir_CL_N**) and the vision potentiometer **PotTargWG**.

$$CmdTurbo\_Tgt = Tgt\_NTurbo\_Base + KaPBaro\_CL\_N + KaTAir\_CL\_N + PotT\arg WG$$

Nota: The CLL channel **CmdTurbo_Tgt** is the same one that is used in case of pressure regulation or turbo speed regulation. So, following the regulation mode (Pressure or Turbo speed), the channel unit is mbar or Rpm.

The following diagram describes the target computation path.



10.4.1.5.4.2 Turbo Speed Error

The close loop regulation error is equal to the turbo speed target described above minus the sensed turbo speed (**N_Turbo**)

$$\text{CmdTurbo\_PID\_Err} = \text{CmdTurbo\_Tgt - N\_Turbo}$$

This value is used as input of the PID function.

Nota: The CLL channel **CmdTurbo_PID_Err** is the same one that is used in case of pressure regulation or turbo speed regulation. So, following the turbo regulation mode, the channel's unit is mbar or Rpm.

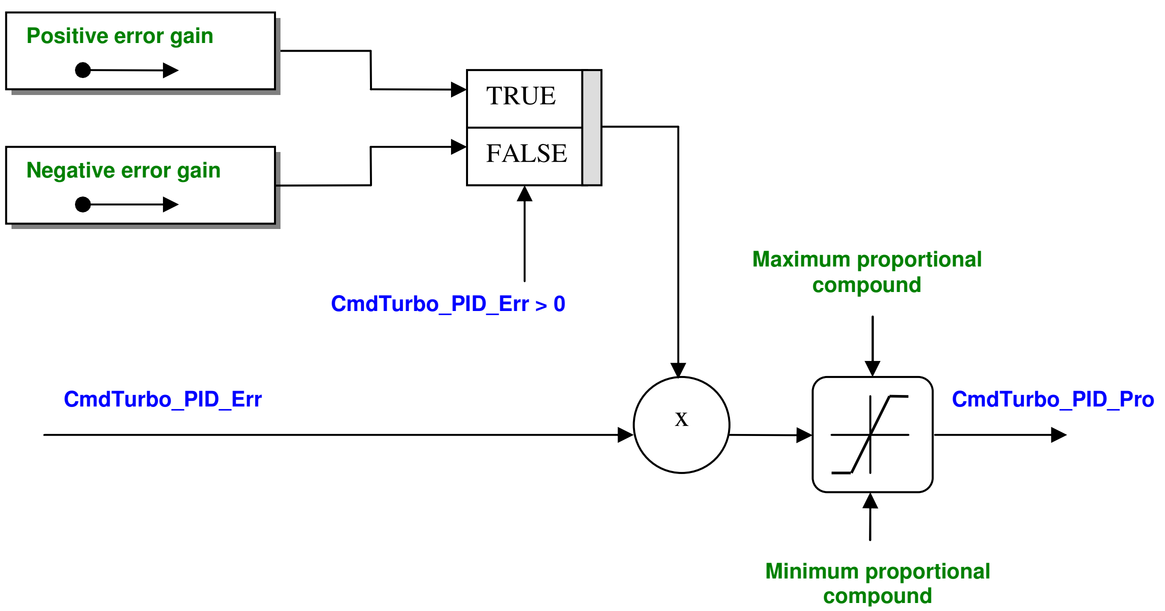
10.4.1.5.5 Turbo PID Function

The regulation processing corresponds to a standard PID regulation. Each regulation type (Turbo Pressure or Turbo Speed) holds its own set of map configuration:

PID Compound			Turbo Pressure Regulation	Turbo Speed Regulation
Proportional compound	Gains	Positive error	EE.TboBf.PRegul.K_pro_pos	EE.TboBf.NRegul.K_pro_pos
		Negative error	EE.TboBf.PRegul.K_pro_neg	EE.TboBf.NRegul.K_pro_neg
	Bounds	Minimum value	EE.TboBf.PRegul.ProTermMin	EE.TboBf.NRegul.ProTermMin
		Maximum value	EE.TboBf.PRegul.ProTermMax	EE.TboBf.NRegul.ProTermMax
Integral compound	Gains	Positive error	EE.TboBf.PRegul.K_int_pos	EE.TboBf.NRegul.K_int_pos
		Negative error	EE.TboBf.PRegul.K_int_neg	EE.TboBf.NRegul.K_int_neg
	Bounds	Minimum value	EE.TboBf.PRegul.IntTermMin	EE.TboBf.NRegul.IntTermMin
		Maximum value	EE.TboBf.PRegul.IntTermMax	EE.TboBf.NRegul.IntTermMax
	Threshold	Minimum error above which the integral is computed	EE.TboBf.PRegul.Int_ErrMin	EE.TboBf.NRegul.Int_ErrMin
		Maximum error below which the integral is computed	EE.TboBf.PRegul.Int_ErrMax	EE.TboBf.NRegul.Int_ErrMax
Derivative compound	Gains	Positive error difference	EE.TboBf.PRegul.Eng_Der_Gain_Pos	EE.TboBf.NRegul.Eng_Der_Gain_Pos
		Negative error difference	EE.TboBf.PRegul.Eng_Der_Gain_Neg	EE.TboBf.NRegul.Eng_Der_Gain_Neg
	Bounds	Minimum value	EE.TboBf.PRegul.DerTermMin	EE.TboBf.NRegul.DerTermMin
		Maximum value	EE.TboBf.PRegul.DerTermMax	EE.TboBf.NRegul.DerTermMax
	Thresholds	Minimum error difference above which the derivative is computed	EE.TboBf.PRegul.Eng_DerTerm_ErrDiff_Pos_Min	EE.TboBf.NRegul.Eng_DerTerm_ErrDiff_Pos_Min
		Minimum error difference below which the derivative is computed	EE.TboBf.PRegul.Eng_DerTerm_ErrDiff_Neg_Max	EE.TboBf.NRegul.Eng_DerTerm_ErrDiff_Neg_Max

Note: As the computation logic is the same one in case of Turbo Speed regulation or Turbo Pressure regulation, the PID is described with map names whom regulation type has been substituted to an X. Thus as example, the proportional positive error gains **EE.TboBf.PRegul.K_pro_pos** and **EE.TboBf.NRegul.K_pro_pos** are substituted to **EE.TboBf.XRegul.K_pro_pos**.

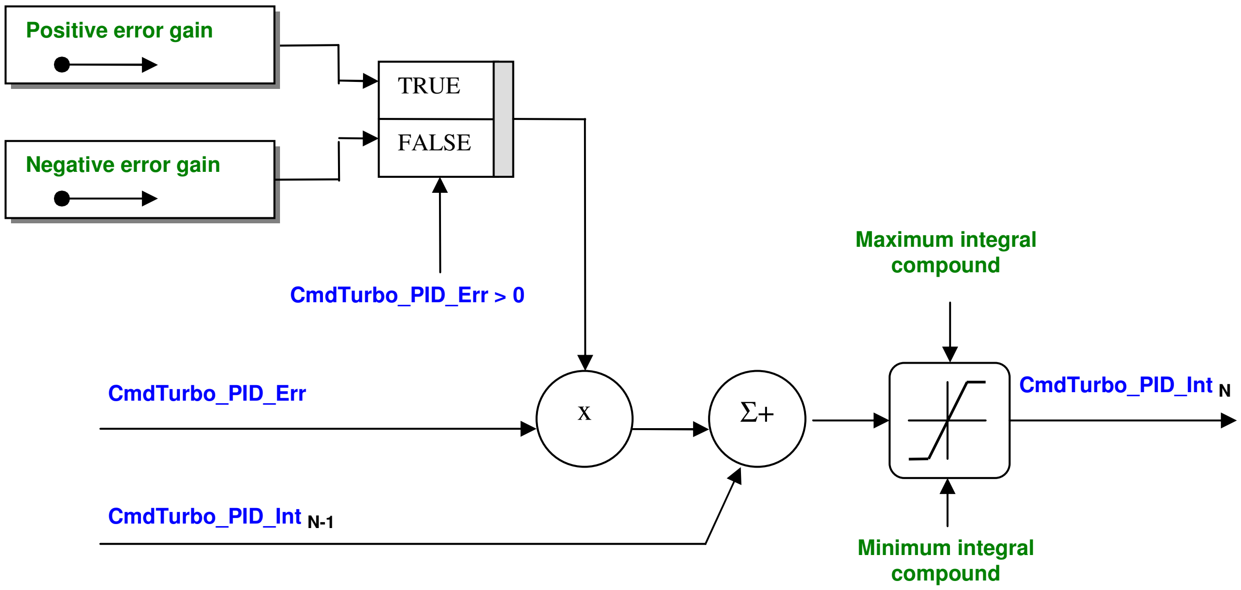
10.4.1.5.5.1 Proportional compound



$$CmdTurbo\_PID\_Pro = CmdTurbo\_PID\_Err * EE.TboBf.XRegul.K\_pro(pos/neg)$$

The turbo proportional term is limited from **EE.TboBf.*X*Regul.ProTermMin** to **EE.TboBf.*X*Regul.ProTermMax** .

10.4.1.5.5.2 Integral compound



The integral term is updated only if the target error is included in the range defined by **EE.TboBf.*X*Regul.Int_ErrMin** and **EE.TboBf.*X*Regul.Int_ErrMax.**

In this case:

$$CmdTurbo\_PID\_Int_{N} = CmdTurbo\_PID\_Int_{N-1} + (CmdTurbo\_PID\_Err * EE.TboBf.K\_int\_(pos/neg))$$

Outside the range [**EE.TboBf.*X*Regul.Int_ErrMin**, **EE.TboBf.*X*Regul.Int_ErrMax**] , the integral term is frozen to its last value

NOTE: The turbo speed integral term is limited from **EE.TboBf.*X*Regul.IntTermMin** to **EE.TboBf.*X*Regul.IntTermMax** .

10.4.1.5.5.3 Derivative compound

We define the derivative error as follow:

$$CmdTurboPIDErDer_N = CmdTurbo\_PID\_Err_N - CmdTurbo\_PID\_Err_{N-1}$$

When **CmdTurboPIDErDer** is greater than **EE.TboBf.*X*Regul.Eng_DerTerm_ErrDiff_Pos_Min** , and lower than **EE.TboBf.*X*Regul.Eng_DerTerm_ErrDiff_Pos_Max** **, ,** the derivative term is:

$$CmdTurbo\_PID\_Der = CmdTurboPIDErDer * EE.TboBf.XRegul.Eng\_Der\_Gain\_Pos$$

Else if **FD.Tbo.DerTerm_ErrDiff** < **EE.TboBf.Eng_DerTerm_ErrDiff_Pos_Max**, then derivative term will be:

$$CmdTurbo\_PID\_Der = CmdTurboPIDErDer * EE.TboBf.NRegul.Eng\_Der\_Gain\_(Pos/Neg)$$

NOTE: The turbo speed derivative term is limited from **EE.TboBf.DerTermMin** to **EE.TboBf.DerTermMax**

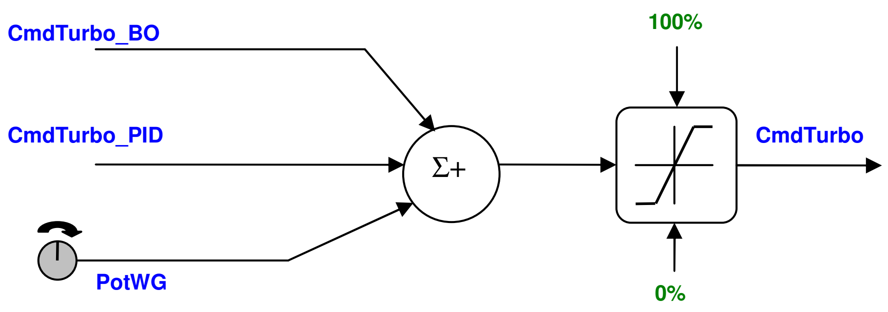
10.4.1.5.6 Final PID output

The final PID term **FD.Tbo.Final_PID** is signed and is the sum of the 3 terms limited from **EE.TboBf.PID_Min** to **EE.TboBf.PID_Max**:

$$CmdTurbo\_PID = CmdTurbo\_PID\_Pro + CmdTurbo\_PID\_Int + CmdTurbo\_PID\_Der$$

10.4.1.6 Final Turbo command

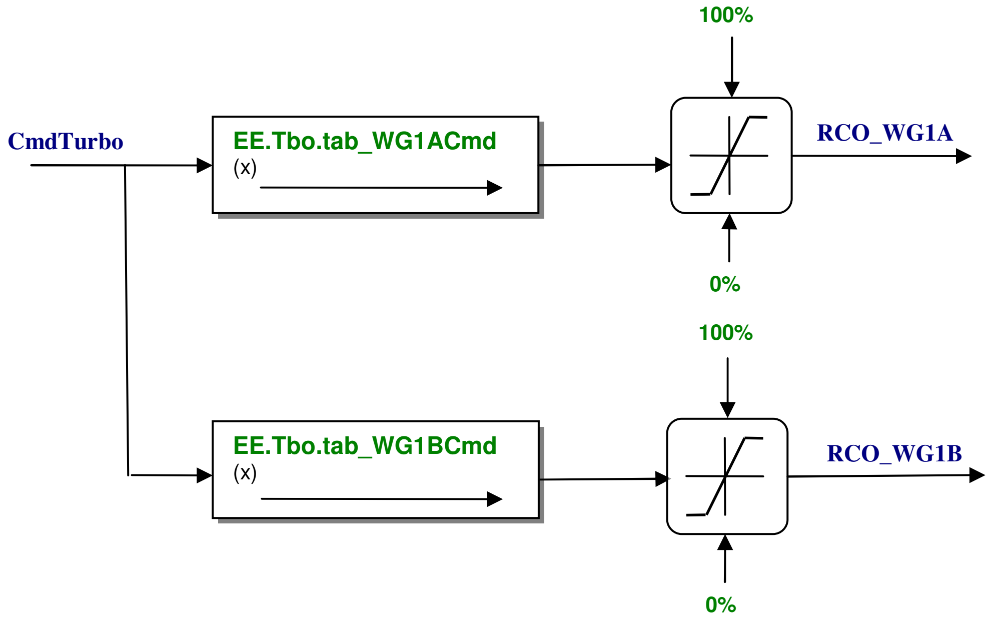
The final turbo command (**CmdTurbo**) is the sum (clipped to [0%, 100%]) of the open loop term plus the PID output, plus the vision potentiometer (**PotWG**).



$$CmdTurbo = CmdTurbo\_BO + CmdTurbo\_PID + PotWG$$

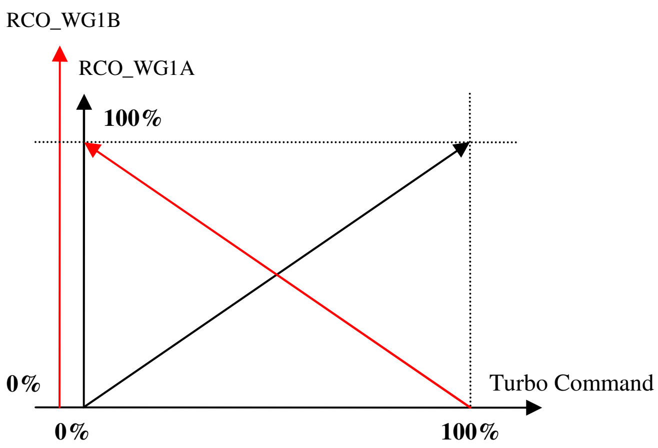
10.4.2 Waste Gates Commands (5 mS)

The waste gates (1A and 1B) PWM commands are function of the turbo command (**CmdTurbo**), following the both user maps **EE.Tbo.tab_WG1ACmd** and **EE.Tbo.tab_WG1BCmd** as a function of the Turbo command (**CmdTurbo**).



Configuration examples

- ✓ **<u>Complementary waste gates: RCO_WG1B = 100% – RCO_WG1A</u>**



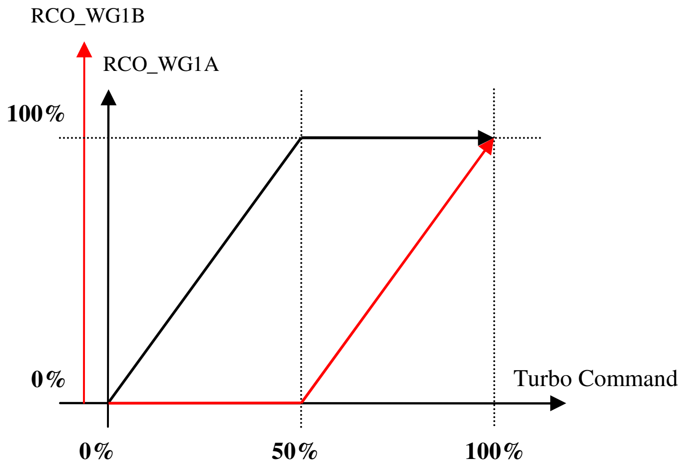
EE.Tbo.tab_WG1ACmd map content

Turbo command Breakpoints (%)	0	13	25	38	50	60	70	80	90	100
Waste Gate 1A RCO command (%)	0	13	25	38	50	60	70	80	90	100

EE.Tbo.tab_WG1BCmd map content

Turbo command Breakpoints (%)	0	13	25	38	50	60	70	80	90	100
Waste Gate 1B RCO command (%)	100	87	75	62	50	40	30	20	10	0

✓ **"Successive" waste gates**



EE.Tbo.tab_WG1ACmd map content

Turbo command Breakpoints (%)	0	13	25	38	50	60	70	80	90	100
Waste Gate 1A RCO command (%)	0	26	50	76	100	100	100	100	100	100

EE.Tbo.tab_WG1BCmd map content

Turbo command Breakpoints (%)	0	13	25	38	50	60	70	80	90	100
Waste Gate 1B RCO command (%)	0	0	0	0	0	20	40	60	80	100

10.4.2.1 Engine Stall

When the engine speed is null (**Regime** = 0), two specific waste gates commands are applied. The waste gate 1A RCO command is set to **EE.Tbo.RcoWG1A_Halt**, and the waste gate 1B RCO command is set to **EE.Tbo.RcoWG1B_Halt**.

10.4.2.2 Turbo Fault

When the turbo fault status is set (failsafe turbo mode), the both waste gates commands are set to **EE.Tbo.RcoWG1A_Fault** and **EE.Tbo.RcoWG1B_Fault**

10.4.2.3 Waste gates test

For test purpose only, the user can apply specific configurable valves commands when the engine speed is null. Following the **Test_WG1** CLL channel value, the waste gate 1A RCO command is set to **EE.Tbo.RcoWG1A_Test** or the waste gates 1B RCO command is set to **EE.Tbo.RcoWG1B_Test**

Valve	**Test_WG1** CLL channel value	Configurable test command (%)
Waste gate 1A	1	**EE.Tbo.RcoWG1A_Test**
Waste gate 1B	2	**EE.Tbo.RcoWG1B_Test**

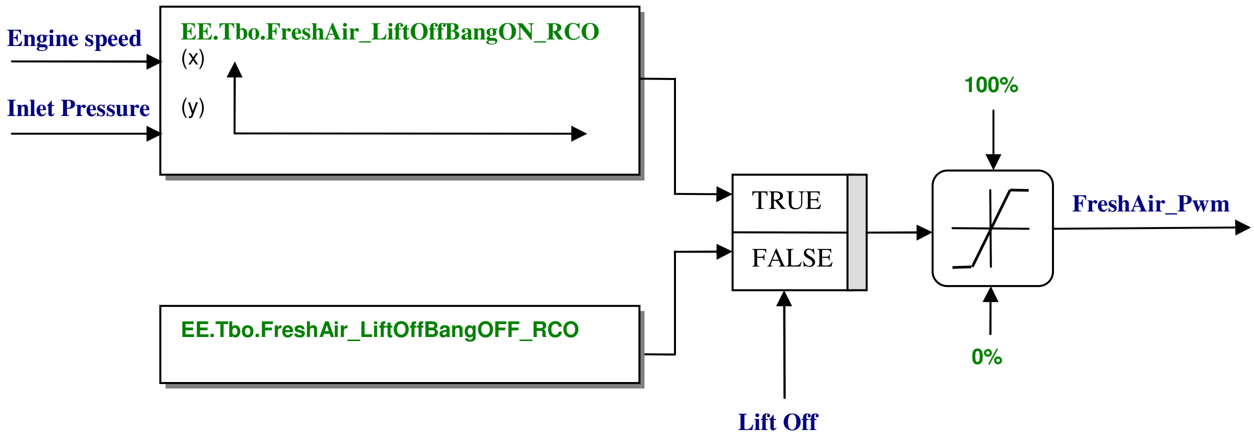
10.4.3 Exhaust Fresh air valve (optional)

In Lift Off Anti-Lag mode, a "fresh air" valve may be commanded (ON/OFF type or PWM type). This option may be activated by selecting an output with the map **EE.CfgU.Freshair_OUT** (see § Outputs selection and assignment in documentation Client_Configuration.doc). The output type (ON/OFF or PWM) is selected via the user map **EE.Tbo.FreshAir_OutType**:

Map content	Output type
0	ON/OFF
1	PWM

10.4.3.1 PWM output

The PWM valve command frequency is set via the user map **EE.Tbo.FreshAir_FreqPwm.**When lift off is active, the RCO command value (clipped to [0%, 100%]) is issued from the map **EE.Tbo.FreshAir_LiftOffBangON_RCO** as a function of engine speed and the inlet pressure. Else, when lift off is inactive, the RCO **EE.Tbo.FreshAir_LiftOffBangOFF_RCO** is applied.



10.4.3.2 ON/OFF output

In that case, the output is simply driven when lift off is active and not driven when lift off is not active.

10.5 Intercooler Spray

10.5.1 Output Selection

In Turbo mode, it is possible to operate intercooler spray.
The output is selected from the map **EE.CfgU.OutputPulverEau (**see § Outputs selection and assignment in documentation Client_Configuration_x.doc)

10.5.2 Command Parameters

The intercooler spray command is a PWM.
The PWM is defined by a period **EE.Tbo.FreqPulvEchangeur** between activations (frequency) and an activation duration **EE.Tbo.RcoPulvEchangeur** in seconds.

10.5.3 Operating Conditions

The intercooler spray command is active if the Perfo switch is in the High position and the air temperature is greater than the threshold **EE.Tbo.STairPulvEchangeur**.