# VEHICLE DYNAMICS OBSERVER COMPONENT

Welcome to VDY!

VDY (VEHICLE DYNAMICS OBSERVER COMPONENT) is a component of the Continental ADAS camera software.

Version info: Version information is found in [vdy\_ver.h](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy__ver_8h.html)

## Table of Contents

1. [Table of Contents](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/index.html#vdy_toc)
2. [Terms and Abbreviations](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_termsAndAbbreviationsPage.html)
   1. [Terms](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_termsAndAbbreviationsPage.html#vdy_termsSection)
   2. [Abbreviations and Acronyms](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_termsAndAbbreviationsPage.html#vdy_abbreviationsSection)
3. [Overview](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_overViewPage.html)
   1. [Component Description](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_overViewPage.html#vdy_overview_CompDescription)
   2. [Inputs and Outputs](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_overViewPage.html#vdy_overview_input_output)
   3. [Component Interdependencies](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_overViewPage.html#vdy_overview_interdependence)
   4. [Sub-Component Interdependencies](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_overViewPage.html#subComponentDependencies)
   5. [OpModes](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_overViewPage.html#vdy_overview_OpModes)
   6. [Requirements](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_overViewPage.html#vdy_overview_requirements)
   7. [Resource Usage](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_overViewPage.html#vdy_overview_resources)
4. [Export Interface](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_exportInterfacePage.html)
   1. [Static View](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_exportInterfacePage.html#vdy_exportInterface_staticView)
      1. [Functions](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_exportInterfacePage.html#vdy_exportInterface_staticView_functions)
      2. [Variables, Types, Constants, Macros](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_exportInterfacePage.html#vdy_exportInterface_staticView_variablesTypesConstantsMacros)
   2. [Dynamic View](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_exportInterfacePage.html#vdy_exportInterface_dynamicView)
      1. [OpModes](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_exportInterfacePage.html#vdy_exportInterface_dynamicView_OpModes)
      2. [Error Codes and DEM Events](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_exportInterfacePage.html#vdy_exportInterface_dynamicView_ErrorCodes)
5. [Configuration](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_configurationPage.html)
6. [Internal Design](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_designPage.html)
   1. [Modules](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_designPage.html#vdy_internalDesign_modules)
      1. [Static Module Interdependencies](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_designPage.html#vdy_internalDesign_modules_interdependenciesStatic)
      2. [Dynamic Module Interdependencies](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_designPage.html#vdy_internalDesign_modules_interdependenciesDynamic)
      3. [VDY Main Module](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYMainModule.html)
      4. [Input Output Monitor](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYMonModule.html)
      5. [SWA - Lateral Acceleration Offset Estimation](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/SWA_LAOffsetEstimation.html)
      6. [Yaw Rate Sensor Signal Processing](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/YWRSensorSignalProcessing.html)
      7. [Wheel Speed Signal Processing](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/WheelSpeedSignalProcessing.html)
      8. [Velocity Correction](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYVelCorr.html)
      9. [Common Functions](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/CommonFunctions.html)
      10. [Yaw Rate Estimation With Lateral Acceleration](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYAye.html)
      11. [Yaw Rate Estimation With Gier Sensor](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYGye.html)
      12. [Yaw Rate Estimation With Steerig Wheel Angle](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYSye.html)
      13. [Yaw Rate Estimation With Wheel Speeds](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYWye.html)
      14. [Yaw Rate Estimation With AYE, GYE, SYE & WYE](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYYe.html)
      15. [Motion State Estimation](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYMotSt.html)
      16. [Wheel Speeds PreProcessing](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYWpp.html)
      17. [Velocity Estimation](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYVe.html)
      18. [Sideslip Angle Estimation](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/VDYSae.html)

Links:

* The Detailed Design for ADAS Sensorics [community](http://connext.conti.de/wikis/home/wiki/Wcacd880f7de3_424a_8158_80436b7178d9?lang=en)
* [Doxygen Documentation](http://www.stack.nl/~dimitri/doxygen/index.html)
* [PlantUML Documentation](http://plantuml.com/sitemap)
* The home page of VDY team: [VDY](http://connext.conti.de/wikis/yourTeamWiki.html)
* The home page of Processing 4 group: [Camera Processing 4 - Wiki](http://connext.conti.de/wikis/home?lang=de-de#!/wiki/Camera%20Processing%204%20-%20Wiki/page/Camera%20Processing%204%20-%20Wiki)
* The home page of sensorics department: [Sensorics Community](http://connext.conti.de/communities/service/html/communitystart?communityUuid=07209aed-4646-46d3-bdad-280e9c2dffae)
* ADAS Development Process: [ADAS Process Map 2.3] <https://processcenter.auto.contiwan.com/pkit/process/setup.do?project_id=37687>

# Terms and Abbreviations

## Terms

|  |  |
| --- | --- |
| **Term** | **Definition** |
| VDY | Vehicle Dynamics |
| SWA | Steering wheel angle (input signal for VDY) |
| NVM | Non Volatile Memory |
| COG | Centre of Gravity |

## Abbreviations and Acronyms

|  |  |
| --- | --- |
| **Abbreviation** | **Meaning** |
| EM | Environment Model - software component that uses yaw rate from VDY to track detected objects |
| EBA | Emergency break assist - software component to emergency breaking |
| ACC | Adaptive Cruise Control |
| LD | Lane Departure |
| NORM | Functions - Frame software that reads the data from sensor on the BUS and then performs the normalization operation and then write the data on the RTE interface so that VDY can read it from there. |

# Overview

## Component Description

VDY is responsible for,

* Estimation of vehicle dynamic signals like Longitudinal Velocity & Acceleration, Yaw rate and Lateral Acceleration by Offset Compensation, Filtering and Plausibilization.
* Vehicle path prediction by providing Curvature estimation & Motion State detection.
* Set the DEMs/Errors in case of
  + Invalid inputs
  + Estimated signal / Variance is crossing threshold
  + Input parameters out of range
  + Input signals out of range
  + NVM data out of range

The VDY Software component estimates the Vehicle DYnamics (VDY) based on Single Track Model. VDY is called every 20ms. Various external/internal input inertial sensoric signals such as steering wheel angle, yaw rate, wheel velocities, lateral acceleration and etc., are processed for the estimation of the vehicle dynamics.   
Because of the availability and the quality of the input signals the VDY component is separated into separate three layers.

1. First layer "Signal processing" is preprocessing the input signals, removes offsets, and supports for each input signal a yaw rate and a corresponding variances.
2. Second layer "State estimation" consist of three modules:
   * The yaw rate fusion module using the preprocessed yaw rates
   * The velocity estimation
   * The motion state estimation
3. The Last layer "Supplementary data" estimates some additional date based on the information from the above layers, like curve, and lateral acceleration.
   * The VDY software component is designed to be flexible towards the infrastructure of the vehicle. Not all input sensors are available in all vehicles, yet any additional source reflecting the dynamic state of the vehicle should be used.

There shall be a module for any potential vehicle dynamics sensor handling the sensor specifics. These modules should be configurable.   
The state variables reflecting the complete dynamic state of the vehicle are the yaw rate, longitudinal velocity and side slip angle. Any other variable, such as curvature, lateral acceleration, longitudinal acceleration, etc. can be derived from these variables.

## Inputs and Outputs

### VDY Component takes several sensor input signals

* + Gier Yaw Rate
    - Angular velocity about the vertical vehicle axis
    - Range: -2.61799 to +2.61799 rad/s
  + Steering Wheel Angle
    - Angular displacement of the steering wheel measured from straight-ahead position.
    - Range: (-1.5708 to +1.5708)\*SWRatio rad; SWRatio is Steering Wheel Ratio parameter
  + Lateral Acceleration
    - Acceleration of vehicle perpendicular to the longitudinal vehicle axis, contains parts of gravity given by transversal slope
    - Range: -15 to +15 m/s^2
  + Four Wheels Speeds (Front-Left, Front-Right, Rear-Left and Rear-Right)
    - Angular velocity of the wheel about its spin axis. Wheel velocity is an equivalent measure derived from wheel speeds in consideration of dynamic circumference.
    - Range: 0 to 115 m/s
  + External Vehicle Velocity (provided by the customer)
    - Vehicle Velocity provided by the customer
    - Range: 0 to 115 m/s
  + External Vehicle Longitudinal Acceleration (provided by the customer)
    - Acceleration of vehicle in direction to the longitudinal vehicle axis, contains parts of gravity given by longitudinal slope
    - Range: -15 to +15 m/s^2
  + Four Wheels Rotation Directions
    - Rotation direction of wheel spin
    - Forward, Reverse or Standstill
  + External Vehicle Direction (provided by the customer)
  + Four wheels Ticks( or Pulse)
    - Accumulated pulses generated by sensor ring connected to revolving wheel. Traveled distance is an equivalent measure derived from wheel pulse counter in consideration of wheel ticks per revolution and dynamic rolling circumference.
    - Range: 1 to 250
  + Gear Position
    - Active gear stage
    - ACT\_GEAR\_POS\_NEUTRAL(0), ACT\_GEAR\_POS\_PARKING(1), ACT\_GEAR\_POS\_FORWARD(2), ACT\_GEAR\_POS\_REVERSE(3)
  + Brake Torque
    - Torque, equivalent to pressure in main brake cylinder
  + Parking Brake
    - Park brake state
    - STAT\_INACTIVE(0), STAT\_ACTIVE(1)

### VDY Vehicle Input Parameters

* + Understeer / Self Steering Gradient
    - Rate of change in steering wheel angle with respect to change in steady-state lateral acceleration.
    - Range: 0.00174533 to 0.00698132 rad s^2/m
  + Steering ratio
    - Rate of change of steering wheel angle at given steering wheel trim position, with respect to change in average steer angel of a pair of steered wheels. In case of active steering systems this parameter is dependent additional vehicle states
    - Range: +1 to +500
  + Wheel base
    - Longitudinal distance between the center of tire contact of pair of wheels on same vehicle side
    - Range: +2 to +4.5 m
  + Track width front/rear
    - Lateral distance between the center of tire contact of pair of wheels on same vehicle axle
    - Range: 1 to 2m
  + Vehicle/curb weight
    - The weight of a fueled automobile with standard equipment but without cargo or passengers
    - Range: 500 to 7500 kg
  + Center of gravity height
    - Imaginary point where the total weight of the body may be thought to be concentrated
    - Range: +0.02 to +1 m
  + Axle load distribution
    - The ratio of the vertical load at rear axle to total vehicle vertical load
    - Range: +0.2 to +0.8
  + Wheel Load Dependency Front/Rear Axle
    - Change of difference in wheel circumferential speed resulting from this load transfer during cornering at front/rear axle
    - Range: 0 to +3.0
  + Wheel Circumference
    - Dynamic wheel rolling circumference
    - Range: 1.5 to 2.5m
  + Wheel ticks per revolution
    - Number of sensor pulses per wheel revolution
    - Range: 1 to 250

### **VDY Output** (with variances)

* + **Longitudinal Dynamics** 
    - Vehicle Velocity
    - Vehicle Acceleration
      * Longitudinal Dnamics are computed based upon either Input Wheel Speed or Wheel Tick Signals (Wheel Pre Processing).
      * If External velocity and acceleration is being provided by the customer, then it can be directly used as an output from VDY.
  + **Lateral Dynamics** 
    - Yaw Rate
    - Lateral Acceleration
    - Vehicle Curvature
    - Driver Intended Curvature
      * Lateral Dynamics are computed based upon Input Yaw Rate, Lateral Acceleration, Steering Wheel Angle and Wheel Speed signals. All of these signals except wheel speeds must be available to compute the above mentioned outputs. Wheel Speed signals if available are used to increase the accuracy of the computed Outputs.
      * Also, it should be known whether these input signals are offset compensated or not. Based upon this info VDY will switch on or off VDY offset computation algorithm.
  + **Motion State** 
    - Motion State is predicted based upon the following inputs
      * Velocity (Either External input or internally computed)
      * Yaw Rate (Internally computed)
      * Wheel Ticks
      * Wheel Direction
      * Brake Torque
      * Gear Position
      * Parking Brake
    - Either Wheel Direction or Gear position must be available. Velocity and Yaw Rate information is must. Other signals increase the accuracy of the computation.
    - If External Motion State or Direction info is being provided by the customer, then it can be directly used as an output from VDY.

The data structures for exchange between vdy and its peer components is defined in the file rte\_type.h which is defined on a global level an included by all components.

The request ports (inputs) and the provide ports (outputs) of the VDY component are listed in [reqVdyPrtList\_t](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/structreq_vdy_prt_list__t.html) and [proVdyPrtList\_t](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/structpro_vdy_prt_list__t.html) respectively, declared in [vdy\_ext.h](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy__ext_8h.html).

|  |
| --- |
| **\_\_\_\_\_VDY Component Input Output \_\_\_\_** |
|  |

|  |  |
| --- | --- |
| **Request\_Ports** | **Provide\_Ports** |
| pBaseCtrlData | pAlgoCompState |
| pNVMRead | pNVMWrite |
| pVehicleInputSignals | pVDYErrors |
| pVehicleParameter | pVDYEstCurves |
|  | pVDYOffsets |
|  | pVehicleDynamicSignals |

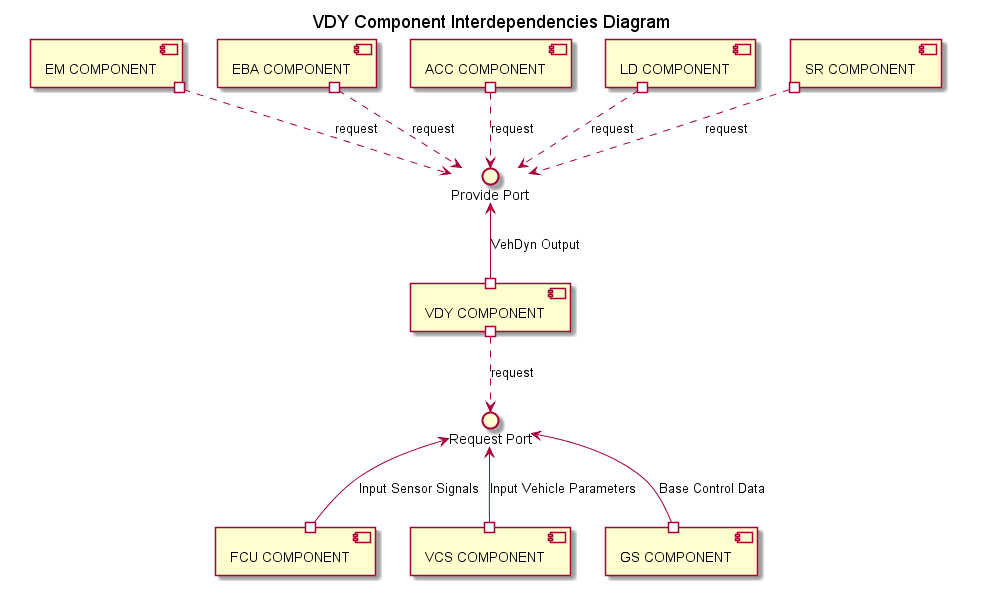
### VDY Component Description Diagram

Diagram

Description automatically generated

### Component Interdependencies

The following figure shows all components VDY has interdependencies with.



### Sub-Component Interdependencies

DOORS link: doors://rbgs854a:40000/?version=2&prodID=0&urn=urn:telelogic::1-503e822e5ec3651e-O-77-00026448

Diagram

Description automatically generated

### OpModes

The VDY component is called in 2 active OpModes.  
1.BASE\_OM\_RESET   
2.BASE\_OM\_RUN   
and Two other OpModes very rarely to be called with are   
3.BASE\_OM\_IDLE   
4.BASE\_OM\_DEMO   
These modes are called by the GS(Global Scheduler) using the eOpMode.

### Requirements

doors://rbgs854a:40000/?version=2&prodID=0&urn=urn:telelogic::1-503e822e5ec3651e-M-000221c8

### Resource Usage

The global scheduling of runtime and memory budgets is maintained in the files [MFC510\_ResBudget](http://ims-adas:7001/si/viewrevision?projectName=%23/ADAS/SW%23Projects/MFC5xx/MFC51x/MFC510/APPL/04_Engineering/03_Workspace/00_Projects/MFC510/resbudget&selection=dpu_budget.xlsm) and [MFC520\_ResBudget](http://ims-adas:7001/si/viewrevision?projectName=%23/ADAS/SW%23Projects/MFC5xx/MFC52x/MFC520/APPL/04_Engineering/03_Workspace/00_Projects/MFC520/resbudget&selection=dpu_budget.xlsm). Therein, the budgets in terms of memory usage and run time (per core) is listed for each OpMode and for each project specific configuration.

# Export Interface

Note

this whole chapter actually describes the component from an architecture point of view. It's an open question how we interlink architecture and design, but we should limit redundancies to a minimum. We welcome your comments on this (and any other) issue on the connext group "Detailed Design for ADAS Sensorics".

## Static View

### Functions

The VDY Component exports one function: [VdyExec()](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/group___v_d_y_main_module_unit.html#ga5ddb7994e9d9f380532068bc84fa9757). All inputs are contained in the [reqVdyPrtList\_t](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/structreq_vdy_prt_list__t.html) Struct reqPorts. This includes an instance of the BaseCtrlData\_t Struct, which has an eOpMode member. This parameter controls which of the OpModes list in [OpModes](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy_exportInterfacePage.html#vdy_exportInterface_dynamicView_OpModes) OpModes is requested.

### Variables, Types, Constants, Macros

Variables, Typedefs, Constants, and Macros that are visible to other components are contained and described in the files [vdy\_cfg\_ext.h](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy__cfg__ext_8h.html), [vdy\_par.h](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy__par_8h.html) and [vdy\_ext.h](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/vdy__ext_8h.html)

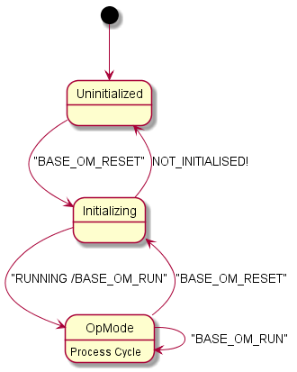
## Dynamic View

OpModes

The VDY Component implements a state machine shown in the following state chart. It consists of one state per OpMode, these are the states in which VDY is actually doing something when [VdyExec()](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/group___v_d_y_main_module_unit.html#ga5ddb7994e9d9f380532068bc84fa9757) is called (with valid parameters), and idle states in between those active states during which VDY is waiting for the next call. Refer to chapter cipp\_overview\_resources and the spreadsheet referenced there for information about the timing behind these function calls.

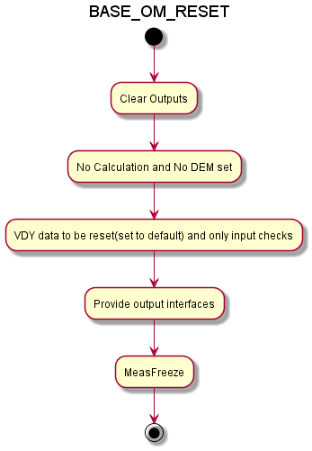
Note

The proposed diagram type to model the Component's life cycle, including the required OpMode order, is a State Chart



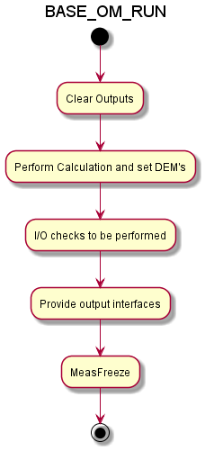
### BASE\_OM\_RESET

If the op-mode sent by the global scheduler is 'BASE\_OM\_RESET' following rules apply: Algo component is called by the scheduler, but no calculation should be done. The algo component must be reset/initialized. This op mode is called in case of ecu startup, sw reset or scheduler mode changes.



### BASE\_OM\_RUN

VDY has only one op mode in which calculations are performed it shall use the default 'BASE\_OM\_RUN'. All derived or specialized running op modes have to fulfill the same requirements.



### Error Codes and DEM Events

VDY has several levels of error/degradation management

* return value of [VdyExec()](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/group___v_d_y_main_module_unit.html#ga5ddb7994e9d9f380532068bc84fa9757) function: The [VdyExec()](mk:@MSITStore:D:\Learning\VDY\Continental_VDY_Design_Documentation_1.8.1.10.chm::/group___v_d_y_main_module_unit.html#ga5ddb7994e9d9f380532068bc84fa9757) function returns   
  BASE\_RETURN\_ERROR if algo execution was not successful.   
  BASE\_RETURN\_OK if eCompState == COMP\_STATE\_RUNNING.
* AlgoCompState: the AlgoCompState\_t provide port contains an CompState\_t element eCompState which follows the specifications rte\_type.h and   
  DOORS Link: doors://rbgs854a:40000/?version=2&prodID=0&view=00000005&urn=urn:telelogic::1-503e822e5ec3651e-O-1639-000221c8
* DEM Events: Under certain conditions, the VDY component sends the DEM Events VDY\_SERVICE\_ERROR, VDY\_FUSI\_ERROR, VDY\_BUS\_DATA\_ERROR, VDY\_INTERNAL\_ERROR as defined in the file Dem\_IntErrIdSlv.h.
* Internal Error Handling: The following are the checks done for handling the internal errors   
  Version number checks, Timestamp checks, NULL checks , Signal status checks, Debounce checks, Data sanity, all Service checks. The following table summarizes all Errors that are used in CIPP:

|  |  |
| --- | --- |
| **Errors** | **Possible Reasons** |
| VDY\_BUS\_DATA\_ERROR | is set if there is no error in CIPP. |
| VDY\_INTERNAL\_ERROR | if the error is not known or not present in the list. |
| VDY\_SERVICE\_ERROR | if the OpMode is invalid/not supported . |
| VDY\_FUSI\_ERROR | is set if the entire input port list is NULL |
| VDY\_FS\_YR\_VS\_WSP | failure w.r.t to Yaw rate monitor check |
| VDY\_FS\_YR\_VS\_AY | failure w.r.t to Yaw rate monitor check |
| VDY\_FS\_YR\_VS\_SWA | failure w.r.t to Yaw rate monitor check |
| VDY\_VEH\_VEL\_NOT\_AVAILABLE | failure w.r.t to vehicle velocity |
| VDY\_YWR\_NOT\_AVAILABLE | failure w.r.t to Yaw rate |
| VDY\_NVM\_LEARN\_DATA\_ERROR | Failure w.r.t NVM read/write |
| VDY\_ERR\_STATE\_UNKNOWN | Unknown state |
| VDY\_ERR\_STATE\_ACTIVE | Any of the internal VDY error is active |
| VDY\_ERR\_STATE\_INACTIVE | No internal VDY error is active |

* VDY contains the logic that decides per OpMode whether the internal error state demands reporting of an error to the frame software by setting CompState\_t eCompState in the AlgoCompState\_t provide port to an error code (COMP\_STATE\_NOT\_INITIALIZED, COMP\_STATE\_RUNNING, COMP\_STATE\_PERMANENT\_ERROR, COMP\_STATE\_TEMPORARY\_ERROR,COMP\_STATE\_SUCCESS,COMP\_STATE\_NOT\_RUNNING ) according to the requirements [DOORS LINK: doors://rbgs854a:40000/?version=2&prodID=0&view=00000005&urn=urn:telelogic::1-503e822e5ec3651e-O-1638-000221c8

The following table summarizes all CompStates that are used in CIPP:

|  |  |
| --- | --- |
| **CompState** | **Possible Reasons** |
| COMP\_STATE\_RUNNING | Must be set in case the algo is called and no error was detected |
| COMP\_STATE\_NOT\_INITIALIZED | An error occurred in OpMode BASE\_OM\_RESET. Refer to ErrorDataList in Measfreeze for details.  VCL is not initialized. |
| COMP\_STATE\_TEMPORARY\_ERROR,COMP\_STATE\_PERMANENT\_ERROR | Should be set in case of errors |
| COMP\_STATE\_SUCCESS | Is reserved for special op modes (e.g. reset, demo, EOL calibration) |
| COMP\_STATE\_NOT\_RUNNING | Shall be set if the algo is called in the idle op mode |