# Module -- Hands on Wheel Detect

# High-Level Description

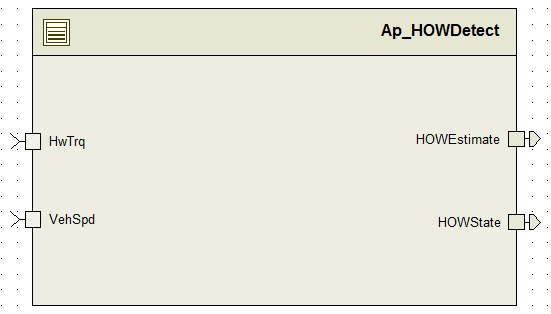
This module is responsible for determining the driver’s hands on the steering wheel. It determines a continuous valued estimate that represents the likelihood that a drivers hands are on the steering wheel (value =1) or off the steering wheel (value=0). A discrete value corresponding to the confidence of the estimate is also specified.

# Figures

## Diagram – Function Data Sharing

No Shared Data

### Component Diagram



# Variable Data Dictionary

For details on module input / output variable, refer to the Data Dictionary for the application. Input / output variable names are listed here for reference.

(Note: Full variable names required in table.)

(Note: All global variables including End Of Line data used should be shown here)

|  |  |  |
| --- | --- | --- |
| Module Inputs | Module Outputs | |
| HwTrq\_HwNm\_f32 | | HOWEstimate\_Uls\_f32 |
| VehSpd\_Kph\_f32 | | HOWState\_Cnt\_s08 |

## Module Internal Variables

This section identifies the name, range and resolutions for module specific data created by this module. If there are no range restrictions on the variable, the term “FULL” is placed into the table for legal range.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Variable Name | Resolution | Legal Range  (min) | Legal Range  (max) | Software Segment |
| HOWDetect\_HOWEstAftRateLimiter\_Uls\_M\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | HOWDETECT\_START\_SEC\_VAR\_CLEARED\_32 |
| HOWDetect\_HOWEstAftLimiter\_Uls\_M\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | HOWDETECT\_START\_SEC\_VAR\_CLEARED\_32 |
| HOWDetect\_AbsHighFreqTrq\_HwNm\_D\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | HOWDETECT\_START\_SEC\_VAR\_CLEARED\_32 |
| HOWDetect\_AbsLowFreqTrq\_HwNm\_D\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | HOWDETECT\_START\_SEC\_VAR\_CLEARED\_32 |
| HOWDetect\_LPF1\_Cnt\_M\_str | Single Precision Float | See Data Dictionary | See Data Dictionary | HOWDETECT\_START\_SEC\_VAR\_CLEARED\_UNSPECIFIED |
| HOWDetect\_LPF2\_Cnt\_M\_str | Single Precision Float | See Data Dictionary | See Data Dictionary | HOWDETECT\_START\_SEC\_VAR\_CLEARED\_UNSPECIFIED |
| HOWDetect\_LPF3\_Cnt\_M\_str | Single Precision Float | See Data Dictionary | See Data Dictionary | HOWDETECT\_START\_SEC\_VAR\_CLEARED\_UNSPECIFIED |
| HOWDetect\_HPF\_Cnt\_M\_str | Single Precision Float | See Data Dictionary | See Data Dictionary | HOWDETECT\_START\_SEC\_VAR\_CLEARED\_UNSPECIFIED |
| HOWDetect\_LPFFinal\_Cnt\_M\_str | Single Precision Float | See Data Dictionary | See Data Dictionary | HOWDETECT\_START\_SEC\_VAR\_CLEARED\_UNSPECIFIED |
| HOWDetect\_LPFEstimate\_Cnt\_M\_str | Single Precision Float | See Data Dictionary | See Data Dictionary | HOWDETECT\_START\_SEC\_VAR\_CLEARED\_UNSPECIFIED |
| HOWDetect\_LPFState\_Cnt\_M\_str | Single Precision Float | See Data Dictionary | See Data Dictionary | HOWDETECT\_START\_SEC\_VAR\_CLEARED\_UNSPECIFIED |

### User defined typedef definition/declaration

This section documents any user types uniquely used for the module.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Typedef Name | Element Name | User Defined Type | Legal Range  (min) | Legal Range  (max) |
| None |  |  |  |  |
|  |  |  |  |  |

# Constant Data Dictionary

## Calibration Constants

This section lists the calibrations used by the module. For details on calibration constants, refer to the Data Dictionary for the application.

|  |
| --- |
| Constant Name |
| t\_HOWVehSpd\_Kph\_u9p7 |
| t\_HOWHighFreqGainY\_Uls\_u9p7 |
| t2\_HOWHFRateX\_HwNm\_u8p8 |
| t2\_HOWHFRateY\_UlspS\_s7p8 |
| t2\_HOWLFRateX\_HwNm\_u8p8 |
| t2\_HOWLFRateY\_UlspS\_s7p8 |
| t\_HOWStateThresholds\_Uls\_u8p8 |
| k\_HOWTrqInitLPFKn\_Hz\_f32 |
| k\_HOWTrqHPFKn\_Hz\_f32 |
| k\_HOWTrqFinalLPFKn\_Hz\_f32 |
| k\_HOWSlewRate\_HwNmpS\_f32 |
| k\_HOWDecaySF\_Uls\_f32 |
| k\_HOWEstLPFKn\_Hz\_f32 |
| k\_HOWMinVehSpd\_Kph\_f32 |
| k\_HOWStateLPFKn\_Hz\_f32 |

## Program(fixed) Constants

### Embedded Constants

All embedded constants whose values are provided in Eng units will be evaluated to the equivalent counts by using the FPM\_InitFixedPoint\_m() macro within the #define statement.

#### Local

|  |  |  |  |
| --- | --- | --- | --- |
| Constant Name | Resolution | Units | Value |
| D\_HOWESTIMATEMAXLIMIT\_ULS\_F32 | Single Precision Float | Uls | 0.5 |
| D\_ABSHFTRQLIMIT\_NM\_F32 | Single Precision Float | Nm | 255.0 |

#### Global

This section lists the global constants used by the module. For details on global constants, refer to the Data Dictionary for the application.

|  |
| --- |
| Constant Name |
| D\_2MS\_SEC\_F32 |
| D\_ONE\_ULS\_F32 |
| D\_ZERO\_ULS\_F32 |
| D\_ZERO\_CNT\_S8 |

### Module specific Lookup Tables Constants

(This is for lookup tables (arrays) with fixed values, same name as other tables)

|  |  |  |  |
| --- | --- | --- | --- |
| Constant Name | Resolution | Value | Software Segment |
| T\_HOWStateValues\_Cnt\_s15p0 | 1 | { -3, -2, -1, 0, 1, 2, 3 } | HOWDETECT\_START\_SEC\_CONST\_16 |

# Functions/Macros used by the Sub-Modules

## Library Functions / Macros

The library and functions / Macros that are called by the various sub modules are identified below,

1. LPF\_Init\_f32\_m
2. HPF\_Init\_f32\_m
3. FPM\_FloatToFixed\_m
4. FPM\_FixedToFloat\_m
5. IntplVarXY\_u16\_u16Xu16Y\_Cnt
6. IntplVarXY\_s16\_u16Xs16Y\_Cnt
7. BilinearXMYM\_s16\_u16XMs16YM\_Cnt
8. TableSize\_m
9. LPF\_OpUpdate\_f32\_m
10. HPF\_OpUpdate\_f32\_m
11. Abs\_f32\_m
12. Limit\_m

## Data Hiding Functions

1. None

## Global Functions/Macros Defined by this Module

### Global Function #1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Function Name** | None | Type | Dir. | Min | Max | UTP Tol. |
| **Arguments Passed** | None |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **Return Value** | None |  |  |  |  |  |

#### Description

None

## Local Functions/Macros Used by this MDD only

### Local Function #1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Function Name** | None | Type | Dir. | Min | Max | UTP Tol. |
| **Arguments Passed** | None |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **Return Value** | None |  |  |  |  |  |

#### Description

None

# Software Module Implementation

## Runtime Environment (RTE) Initial Values

This section lists the initial values of data written by this module but controlled by the RTE. After RTE initialization, the data in this table will contain these values.

|  |  |
| --- | --- |
| Data | Value |
| HOWEstimate\_Uls\_T\_f32 | 0 |
| HOWState\_Cnt\_T\_s08 | 0 |

## Initialization Functions

### Init: HOWDetect\_Init1



#### Design Rationale

None

#### Module Outputs

None

#### Module Internal

None

## Periodic Functions

### Per: HOWDetect\_Per1()

#### Design Rationale

None

#### Program Flow Start

Rte\_Call\_HOWDetect\_Per1\_CP0\_CheckpointReached()

#### Store Module Inputs to Local copies

HwTrq\_HwNm\_T\_f32 = Rte\_IRead\_HOWDetect\_Per1\_HwTrq\_HwNm\_f32()

VehSpd\_Kph\_T\_f32 = Rte\_IRead\_HOWDetect\_Per1\_VehSpd\_Kph\_f32()

#### (Processing of function)





#### Store Local copy of outputs into Module Outputs

Rte\_IWrite\_HOWDetect\_Per1\_HOWEstimate\_Uls\_f32(HOWEstimate\_Uls\_T\_f32)

Rte\_IWrite\_HOWDetect\_Per1\_HOWState\_Cnt\_s08(HOWState\_Cnt\_T\_s08)

#### Program Flow End

Rte\_Call\_HOWDetect\_Per1\_CP1\_CheckpointReached()

## Fault Recovery Functions

None

### FaultRec:

None

#### Design Rationale

None

#### Program Flow Start

None

#### Store Module Inputs to Local copies

None

#### Processing of function

None

#### Store Local copy of outputs into Module Outputs

None

#### Program Flow End

None

## Shutdown Functions

None

### Shtdn:

None

#### Design Rationale

None

#### Program Flow Start

None

#### Store Module Inputs to Local copies

None

#### Processing of function

None

#### Store Local copy of outputs into Module Outputs

None

#### Program Flow End

None

## Interrupt Functions

None

### Isr:

None

#### Design Rationale

None

#### (Processing of the ISR function)

None

## Serial Communication Functions

None

### SComm:

None

#### Design Rationale

None

#### Program Flow Start

None

#### Store Module Inputs to Local copies

None

#### (Processing of function)

None

#### Store Local copy of outputs into Module Outputs

None

#### Program Flow End

None

# Execution Requirements

## Execution Sequence of the Module

(Describe in words relevant details about the execution sequence of the different sub modules.)

## Execution Rates for sub-modules called by the Scheduler

This table serves as reference for the Scheduler design

|  |  |  |
| --- | --- | --- |
| Function Name | Calling Frequency | System State(s) in which the function is called |
| HOWDetect\_Init1 | On Init | On Init |
| HOWDetect\_Per1 | 2ms | All |

## Execution Requirements for Serial Communication Functions

|  |  |
| --- | --- |
| Function Name | Sub-Module called by (Serial Comm Function Name) |
| None |  |

# Memory Map Definition Requirements

## Sub Modules (Functions)

This table identifies the software segments for functions identified in this module.

|  |  |
| --- | --- |
| Name of Sub Module | Software Segment |
| HOWDetect\_Init1 | RTE\_START\_SEC\_AP\_HOWDETECT\_APPL\_CODE |
| HOWDetect\_Per1 | RTE\_START\_SEC\_AP\_HOWDETECT\_APPL\_CODE |

## Local Functions

This table identifies the software segments for local functions identified in this module.

|  |  |
| --- | --- |
| Name of Sub Module | Software Segment |
| None |  |

# Known Issues / Limitations With Design

1. None

# Unit Test Considerations

The inputs of the high pass filter (**HOWDetect\_HPF\_Cnt\_M\_str.LPF.K** and **HOWDetect\_HPF\_Cnt\_M\_str.CF**) should not be chosen randomly. Because these two inputs for HPF should be generated by HPF init macro (used in function HOWDetect\_Init1). One possible approach is when testing the HOWDetect\_Per1, values for **HOWDetect\_HPF\_Cnt\_M\_str.LPF.K** and **HOWDetect\_HPF\_Cnt\_M\_str.CF** should only use pairs of values from the output of the HOWDetect\_Init1 function. That is, for any one test vector for HOWDetect\_Per1, the values for **HOWDetect\_HPF\_Cnt\_M\_str.LPF.K** and **HOWDetect\_HPF\_Cnt\_M\_str.CF** should be from the output of any one test vector of the HOWDetect\_Init1 function.

# Revision Control Log

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item #** | **Rev #** | **Change Description** | **Date** | **Author Initials** |
| 1 | 1 | Initial component creation | 17-April-14 | Rijvi |
| 2 | 2 | Added unit test considerations | 31-Oct-14 | Rijvi |