NIT

AUTOMOTIVE SOFTWARE WITH AUTOSAR

AUTOSAR

Agenda

- **Diagnostics**
- CANTP - CAN Transport Layer
- DCM - Diagnostic Communication Manager
- DEM - Diagnostics Event Manager
- FIM - Function Inhibition Manager
- Exercise 6 - Diagnostics

Diagnostics
AUTOSAR BSW Modules for Diagnostics

Diagnostics provided by

› DCM

› DEM

› FIM

Required modules

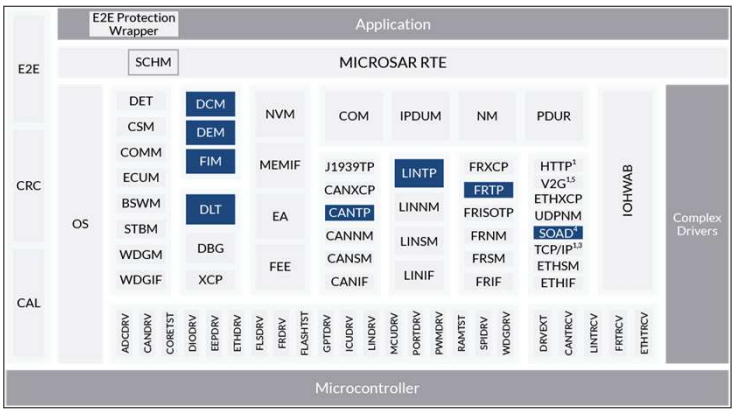
› CANTP

› FRTP

› LINTP

› SOAD (DoIP)

› Diagnostic Log and
  › DLT uses DCM, DEM, DET



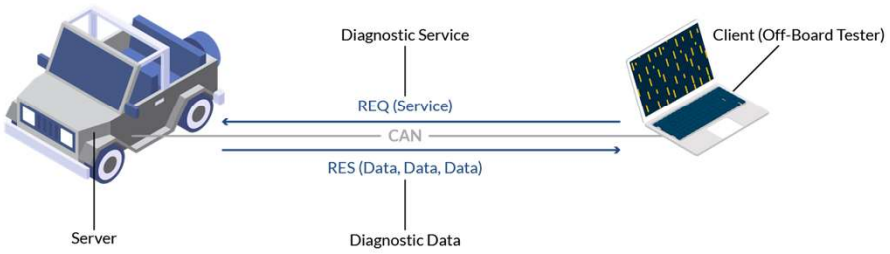
Diagnostics

ISO 14229-1

- Also known as: UDS = Unified Diagnostic Services
- Definition of diagnostic protocol with services
- Network-independent
- Definitions
  - A "diagnostic protocol" is necessary for communication between a diagnostic tester and an ECU
  - A diagnostic protocol incorporates a set of diagnostic services that are transmitted serially between the diagnostic tester and ECU(s)
  - A diagnostic protocol contains data and information on the meanings of the messages to be sent and the ECU's behavior in response to a message

Diagnostics

Network View of Diagnostics



5

- CONFIDENTIAL MATERIAL -

Diagnostic Request is either **functional** or **physical.** This terms refer to the addressing strategy. Functional addressing is a "broadcast" for all ECUs; physical addressing is peer-to-peer communication (point-to-point).

- **A physical request** will be answered **physically** as this accounts for the peer-to-peer property of the addressing.
- **A functional request** will also be answered **physically,** since the tester has to differentiate the responses of all ECUs from each other.

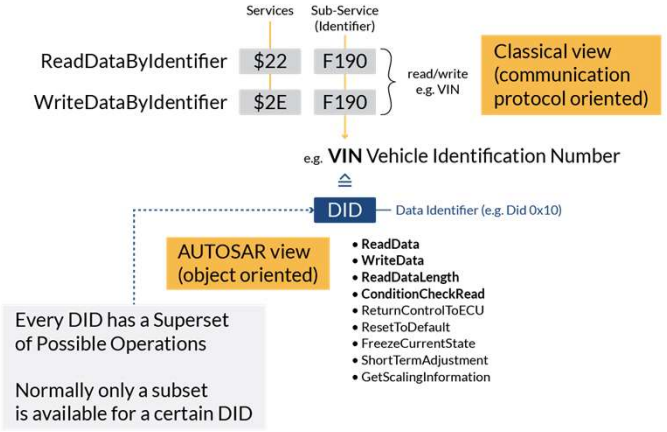
Diagnostic Requests (REQ) and **Diagnostic Responses (RES)** may already contain data, but this is not necessarily the case and depends on the definition of the diagnostic service. A diagnostic request will typically be followed by a diagnostic response. If there is no response at all the tester will observe a timeout. Some services, like the functional service Tester Present, do not need any response from the ECU at all.

The **Diagnostic Response (RES)** can be one of the following:

- **Positive Response:** The service request has been processed correctly and a response, possibly with data, is on its way to the tester.
- **Negative Response with Negative Response Code (NRC):** The service processing encountered an error specified by the NRC code, and this is indicated to the tester.
- **Negative Response with NRC RequestCorrectlyReceiveResponsePending (RCRRP):** The diagnostic server needs more time to process the request and can not yet deliver a positive or negative response. According to UDS a diagnostic server can respond with RCRRP indefinitely. However, all services are expected to be processed within a finite time. The protocol is not finished unless the ECU answers with a final positive or negative response.

Diagnostics

DID – Data Identifier



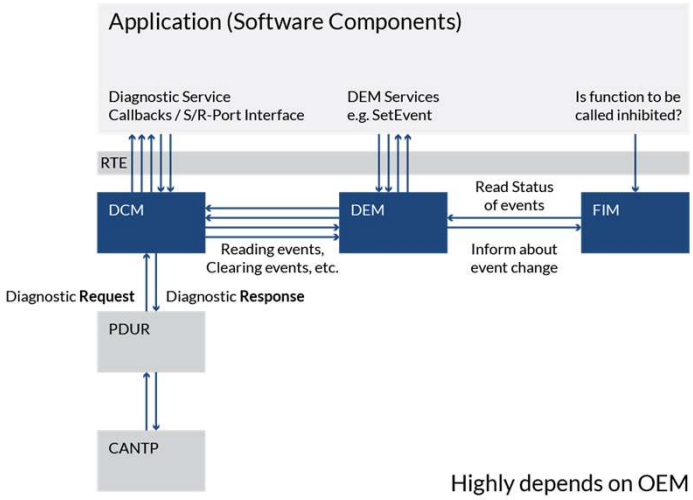
6 - CONFIDENTIAL MATERIAL -

In AUTOSAR configuration only DIDs are available.

A DID is "translated" into UDS. Every Operation on a specific DID is translated into a Service and Service Identifier, i.e., the appearance on the bus is the same as non-AUTOSAR UDS.

Diagnostics

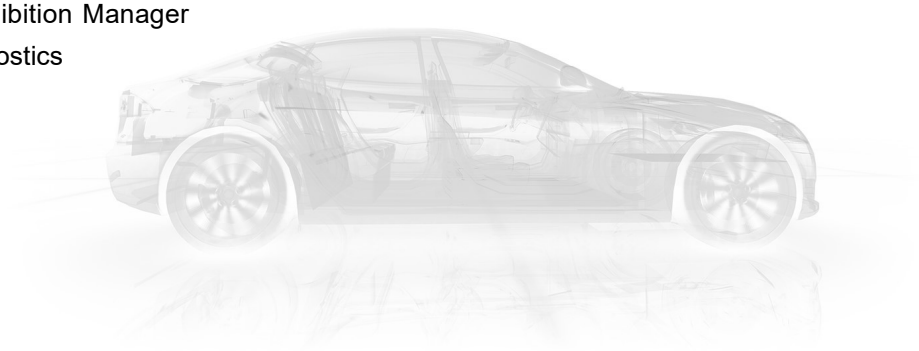
DCM – DEM - FIM



Highly depends on OEM

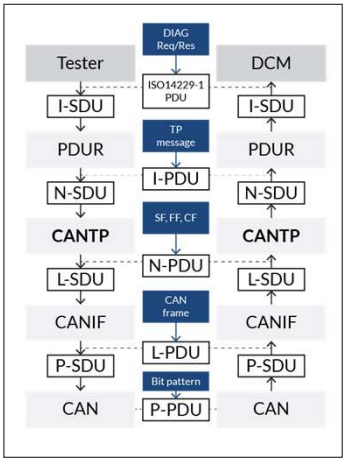
Agenda

- Diagnostics
- **CANTP - CAN Transport Layer**
- DCM - Diagnostic Communication Manager
- DEM - Diagnostics Event Manager
- FIM - Function Inhibition Manager
- Exercise 6 - Diagnostics



CANTP – CAN Transport Layer
Module short information

- CANTP is the module between the PDU Router and the CAN Interface module
- The main purpose of the CAN TP module is to segment and reassemble diagnostic I-PDUs longer than 8 bytes
  - A TP message is called an N-SDU (Network-layer Service Data Unit)
  - Handling of Single (SF), First (FF) and Consecutive Frames (CF) uses N-PDUs (L-SDUs)
- Support of different addressing formats
  - Functional and physical
  - Normal, extended, and mixed



9 - CONFIDENTIAL MATERIAL -

N_SA	Network Source Address
N_TA	Network Target Address
N_TAtype	Network Target Address Type
Mtype	Message Type diagnostics, remote diagnostics
N_PCI	Network Layer Protocol Information
N_data	Payload

Normal addressing format
A unique CAN identifier is assigned to each combination of N_SA, N_TA, N_TAtype and Mtype. N_PCI and N_Data are filed in the CAN frame data.

Extended addressing format
A unique CAN identifier is assigned to each combination of N_SA and Mtype. A unique address is filed to each combination of N_TA and N_TAtype in the first data byte of the CAN frame data field. NPCI and N_Data are filed in the remaining bytes of the CAN frame data field.

Mixed addressing format
A unique CAN identifier is assigned to each combination of N_SA, N_TA, N_TAtype. N_AE is placed in the first data byte of the CAN frame data field. N_PCI and N_Data are placed in the remaining bytes of the CAN frame data field.

Agenda

- Diagnostics
- CANTP - CAN Transport Layer
- **DCM - Diagnostic Communication Manager**
- DEM - Diagnostics Event Manager
- FIM - Function Inhibition Manager
- Exercise 6 - Diagnostics

DCM – Diagnostic Communication Manager
Overview

- DCM is network independent
- Provides a common API for diagnostic services
- The functionality of this AUTOSAR-Basic-SW is used by external diagnostic tools e.g., in the development, manufacturing, service or legislative OBD
- The DCM handles different (application layer) **diagnostic protocols**
  - **OBD** (ISO 15031-5)
  - Enhanced diagnostics (ISO 14229-1)
- For diagnostic **session handling** the network independent sections of the following specifications are also handled by the DCM
  - ISO 15765-3: Implementation of unified diagnostic services (UDS on CAN)
  - ISO 15765-4: Requirements for emissions-related systems

DCM – Diagnostic Communication Manager
Overview

OBD (On Board Diagnostics) refers to legislative diagnostics requirements defined by the respective country in which the vehicle is on the road. OBD is related to the so called "emission related" diagnostics.

In OBD, the diagnostic tester is often referred to as the "generic scan tool."

Among other things, OBD requires:

› Parallel operation to the "ordinary" UDS diagnostic communication

› Special UDS services with their own semantics defined for OBD diagnostics only

Examples:

› DCM: OBD communication services $01 - $0A (defined in SAE J1979 Rev May 2007)

› DEM: a special OBD FreezeFrame "record 0" etc.

DCM – Diagnostic Communication Manager
Overview

Implementing the above requirements in DEM and DCM, AUTOSAR OBD functionality meets all **light duty OBD regulations** in the world

- EOBD, Japan OBD, California OBDII,
- Open issues, not treated in AUTOSAR
- **MIL** (Malfunction Indicator Lamp): blinking, light bulb check of the MIL, etc
- Misfire fault handling: Debouncing and filtering
- No description on how to achieve OBD compliant diagnostic applications concerning state handling or diagnostic algorithms.
- This is due to the difficulties that OBD regulations may change frequently as they are driven by politics and differences over the nations.

DCM – Diagnostic Communication Manager
Overview

The DCM

- Ensures diagnostic data flow
- Manages the diagnostic states
  - Diagnostic sessions
  - Security states
- Checks that the diagnostic service request is supported
- Checks that the service may be executed in the current session and with the current security state

DCM – Diagnostic Communication Manager
Overview

- In the AUTOSAR architecture, the Diagnostic Communication Manager (DCM) is located in the Communication Services (Service Layer)
- In the Communication process, the DCM receives a diagnostic message from the PDU Router
- The DCM will check the diagnostic message
- Depending on the Diagnostic Service ID (SID), the corresponding calls in the Application Layer will be done
- Supported services
  - UDS
  - OBD
  - Flash Bootloader Interaction
    - Jump from App ←→ FBL, Flag Management

DCM – Diagnostic Communication Manager
UDS services

Service Id	UDS Service
0x23	ReadMemoryByAddress (RMBA)
0x28	CommunicationControl
0x2C	DynamicallyDefineDataIdentifier
0x31	Remote Activation of Routine
0x34	Request Download
0x35	Request Upload
0x36	DataTransfer
0x37	RequestTransferExit
0x3D	WriteMemoryByAddress (WMBA)
0x86	ResponseOnEvent (RoE)
…	…

Example

Diagnostic job name: Internal light control

Diagnostic job Id = 0xBA80

Diagnostic job Id argument:

0x00 – Turn off light

0x01 – Turn on light

[Source Target PayloadLength] [Payload]

Request

[0xA2|0x20|0x00|0x05] [0x31|0x01|0xBA|0x80|0x01]

Negative response

[0x20|0xA2|0x00|0x03] [0x7F|0x31|0x31]

Positive response

[0x20|0xA2|0x00|0x05] [0x71|0x01|0xBA|0x80|0x01]

16 - CONFIDENTIAL MATERIAL -

Callouts for RMBA/WMBA

- `Dcm_ReadMemory`
- `Dcm_WriteMemory`

The AUTOSAR architecture doesn't provide the possibility to access the ECU memory using a physical address. This is implemented using a BlockId which identifies a memory block. Due to this fact, the DCM is not able to fully support the implementation of IS014229-1 services which use a physical memory access.

As a solution, the DCM defines a callout to implement this kind of memory access. This callout implementation could simply be realized by defining a mapping between the BlockId and the physical memory address.

Further callouts of the DCM:

- `Dcm_Confirmation` - successful transmission or error during execution of a diagnostic service.
- `Dcm_SetProgConditions` - store Bootloader related information before jump into Bootloader.
- `Dcm_GetProgrConditions` - read Bootloader information upon startup of ECU (`Dcm_Init()`) to determine if a RC 0x50 or 0x51 has to be sent.
- `Dcm_ProcessRequestTransferExit` - DCM calls this if a download or upload shall be exited.
- `Dcm_ProcessRequestUpload` - start upload
- `Dcm_ProcessRequestDownload` - start download

DCM – Diagnostic Communication Manager
OBD services

Service Id	OBD Service
0x01	Request Current Powertrain Diagnostic Data
0x02	Request Power Train FreezeFrame Data
0x03	Request Emission-Related Diagnostic Trouble Codes
0x04	Clear/Reset Emission-Related Diagnostic Information
0x06	Request On-Board Monitoring Test-results for Specific Monitored System
0x07	Request Emission Related Diagnostic Trouble Codes Detected during Current or Last Completed Driving Cycle
0x08	Request Control of On-Board System, Test or Component
0x09	Request Vehicle Information
0x0A	Request Emission Related Diagnostic Trouble Codes Detected with Permanent Status

DCM – Diagnostic Communication Manager
DCM ComM Request

- DCM requests and releases communication directly through the ComM
  - During an ongoing communication service
    - DCM calls `ComM_DCM_ActiveDiagnostic` upon reception of the UDS request (`Dcm_TpRxIndication`)
    - DCM calls `ComM_DCM_InactiveDiagnostic` upon transmission of the UDS response (`Dcm_TpTxconfirmation`)
  - Upon transition into a non-default session
    - `ComM_DCM_ActiveDiagnostic`
  - Upon transition into the default session
    - `ComM_DCM_InactiveDiagnostic`
- Communication Control service is an exception; this is handled by BswM.
- Additionally DCM notifies the BSWM about its current Mode using the `BswM_DCM` API.

DCM – Diagnostic Communication Manager
Mode Management

- The DCM sends mode requests to the BswM based on the UDS requests it receives
- DCM acts as a Mode manager for
  - DcmDiagnosticSessionControl (service 0x10)
  - DcmEcuReset (part of service 0x11)
  - DcmModeRapidPowerShutDown (part of service 0x11).
  - DcmCommunicationControl_<symbolic name of ComMChannelId> (service 0x28)
  - DcmControlDTCSetting (service 0x85)
  - DcmResponseOnEvent_<RoeEventID> (service 0x86)

Example: DCM can request "Disable Normal Communication.“During this mode BswM will turn off the corresponding I-PDU groups and NM PDUs.

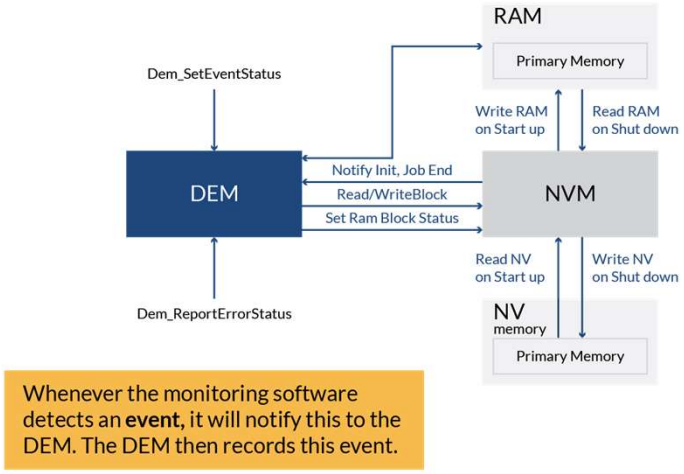
DCM – Diagnostic Communication Manager
Data use port

- Client / Server communication can be used for DIDs
  - Synchronous or asynchronous handling supported
  - DID Operations are implemented as Server Runnables in the application SWCs
- Sender / Receiver communication can also be used for DIDs
  - P-Port Interface for DcmDspDidRead
  - P-Port Interface for DcmDspDidWrite
- Scaling at Ports
  - The CompuMethod for the data type can optionally be derived from the DcmDspDiagnosisScaling

Agenda

- Diagnostics
- CANTP - CAN Transport Layer
- DCM - Diagnostic Communication Manager
- **DEM - Diagnostics Event Manager**
- FIM - Function Inhibition Manager
- Exercise 6 - Diagnostics

DEM – Diagnostic Event Manager
Module operation



Whenever the monitoring software detects an **event**, it will notify this to the DEM. The DEM then records this event.

22 - CONFIDENTIAL MATERIAL -

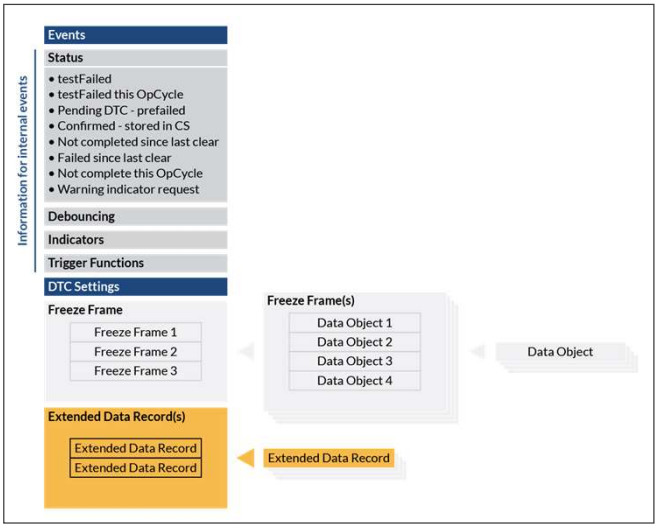
SWCs normally signal fault Events to the DEM by invocation of the `Dem_SetEventStatus()` API. However not only SWCs can report Events to the DEM. The Basic Software Modules can also set "internal" Events by using the `Dem_ReportErrorStatus()` API

The DEM uses various NvM Blocks to store its data:
- **AdminDataBlock:** timestamps and operation cycle counter are saved every driving cycle
- **StatusDataBlock:** Event data
- **DemPrimaryDataBlockO..n:** The number of remembered DTCs (primary or secondary).

The **DEM** triggers NVM Block operations during runtime, but some blocks like StatusData and AdminData are only marked as valid and changed with `NvM_SetRamBlockStatus` API and actually written upon shutdown in `NvM_WriteAll()`.

DEM – Diagnostic Event Manager
Events

- Events have a status
- Learning" and Forgetting" faults: (pre-) debouncing
- Indicator
- DTC information
- Additional information to specify an event
- Freeze Frames
  - Freeze frames have data objects
- Extended data records



23 - CONFIDENTIAL MATERIAL -

Freeze Frame:
OEM-specific set of the vehicle/system operation conditions at a specific time. The term "Freeze Frame" is defined as a record of data (DIDs / PIDs). ISO 14229-1 refers to FreezeFrames as "SnapShotRecords". There is the possibility to use pre-stored FreezeFrames. It means the data is already available and does not need to be acquired when the event status changes. This way time-critical Freeze Frame data can be saved together with the event.

Extended data records:
OEM-specific additional information assigned for a distinctive event which is not contained in the Freeze Frame and also described in ISO 14229-1. Extended Data Records are implemented by callbacks to the application. These callbacks are issued by the DEM at the point in time when the Diagnostic Monitor sets the error event in the DEM. The diagnostic configuration can provide 1..n extended data records for a DEM event. Depending on the point in time, different extended data records will be recorded together with the error event. In this way the first occurrence and the last occurrence can be considered.

Example for data not contained in Freeze Frames are: statistical information like **most recent** occurrence counter and **most recent** operation cycle / aging counter (they change over time). The same information contained in Freeze Frames would not change and is **"frozen".**

DTC Information
Diagnostic Trouble Code is a numeric code which consists of 3 bytes unique identifier for the error event. Together with this information, the status byte is also always saved. That means the minimum memory consumption of a DTC is 4 bytes.

DEM – Diagnostic Event Manager

Event Storage

- Each DTC (Diagnostic Trouble Code) is coupled with one byte providing status information:
  - testFailed (bit0)
  - testFailedThisOperationCycle (bit1)
  - pendingDTC - prefailed (bit2)
  - confirmedDTC - stored in Primary Memory (bit3)
  - testNotCompletedSinceLastClear (bit4)
  - testFailedSinceLastClear (bit5)
  - notCompletedThisOperationCycie (bit6)
  - warningIndicatorRequest (bit7)

testFailed (bit0)
If the testFailed bit is set it means that the fault is active. The test for an error condition or wrong behavior has succeeded.

testFailedThisOperationCycle (bits)
This bit is only set if the fault occurs in the current operation cycle. If the fault is active and this flag is not set it means that the fault occurred in a previous operation cycle.

pendingDTC - prefailed (bit2)
Indicates that the corresponding DTC will be set in the near future.

confirmedDTC - stored in Primary Memory (bit3)
As soon as the fault is stored in the Primary Memory, this bit is set. If the fault is cleared from the event storage the value of this bit will be changed too.

testNotCompletedSinceLastClear (bit4)
If there is no information about whether the DTC test has passed or failed since the last clear, this flag is active.

testFailedSinceLastClear (bit5)
This flag informs you that the DTC has set since the last clearing action.

notCompletedThisOperationCycie (bit6)
This bit corresponds with the NotCompletedSinceLastClear. It informs whether the complete information was set or reset in the current operation cycle.

warningIndicatorRequest (bit7)
Indicates whether the occurrence of this DTC should set a warning indicator.

DEM – Diagnostic Event Manager
Event Storage

- To provide the tester with more information about the point in time a DTC occurred, additional information is stored. This information is OEM-specific and is split into standardized and optional data.
- Examples for OEM-specific standardized data:
  - **Occurrence Flag**
    - Distinguishes an entry as occurrence or fault
  - **Original Odometer**
    - This is the odometer value when the fault occurred for the first time.
  - **Most Recent Odometer**
    - This is the odometer value when the fault occurred the most recent time.
  - **Frequency**
    - This is the total number of times the DTC status has transitioned from "stored active" to "active."

DEM – Diagnostic Event Manager

Event Status

An event status can be reported to the DEM via API

```
Dem_SetEventStatus(Dem_EventIdType EventId, uint8 EventStatus)
```

- Argument values of `EventStatus`:
  - `DEM_EVENT_STATUS_PASSED`: No debouncing, event is set to passed.
  - `DEM_EVENT_STATUS_FAILED`: No debouncing, event is set to failed.
  - `DEM_EVENT_STATUS_PREPASSED`: Debouncing relevant, configured debouncing done.
  - `DEM_EVENT_STATUS_PREFAILED`: Debouncing relevant, configured debouncing done.

DEM – Diagnostic Event Manager
Internal and external events

- EventKind **BSW events**
  - Cannot be queried over UDS
  - The BSW reports them as Production Errors
  - Defined by AUTOSAR
  - Can be used / redefined as EventKind SWC (e.g., CANSM)
- EventKind **SWC events**
  - Defined by the application
  - Can be queried over UDS

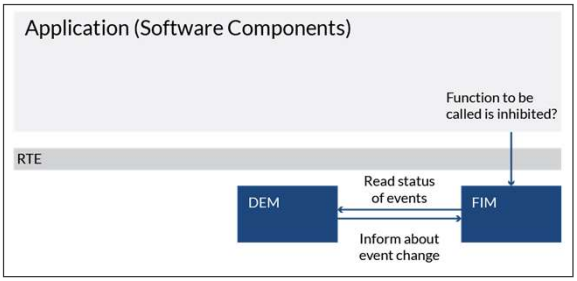
Agenda

- Diagnostics
- CANTP - CAN Transport Layer
- DCM - Diagnostic Communication Manager
- DEM - Diagnostics Event Manager
- **FIM - Function Inhibition Manager**
- Exercise 6 - Diagnostics

FIM – Function Inhibition Manager
Overview

The FIM knows which functions are inhibited at the moment

- The application has to ask the FIM whether a certain function can be run or not.
  - FIM either stores the condition for each function identifier (FID) and is triggered by DEM on occurrence of an event
  - Or FIM requests the current events state from DEM and evaluates the condition for function identifier on request.



Agenda

- Diagnostics
- CANTP - CAN Transport Layer
- DCM - Diagnostic Communication Manager
- DEM - Diagnostics Event Manager
- FIM - Function Inhibition Manager
- **Exercise 6 - Diagnostics**