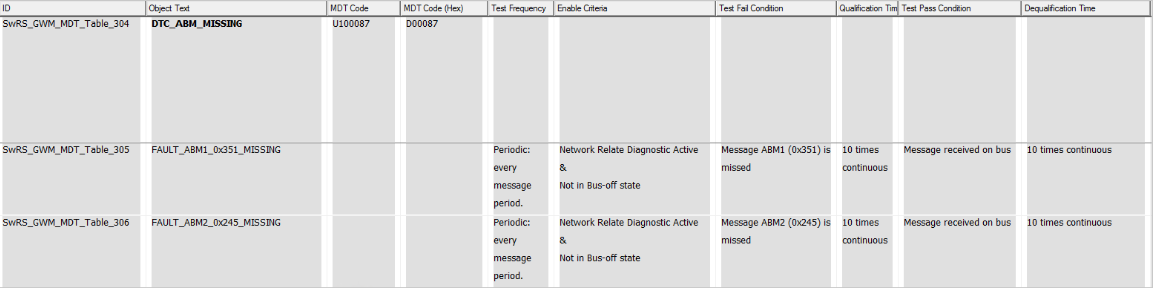
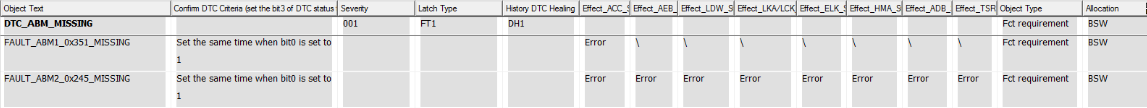
Check DOORS requirement

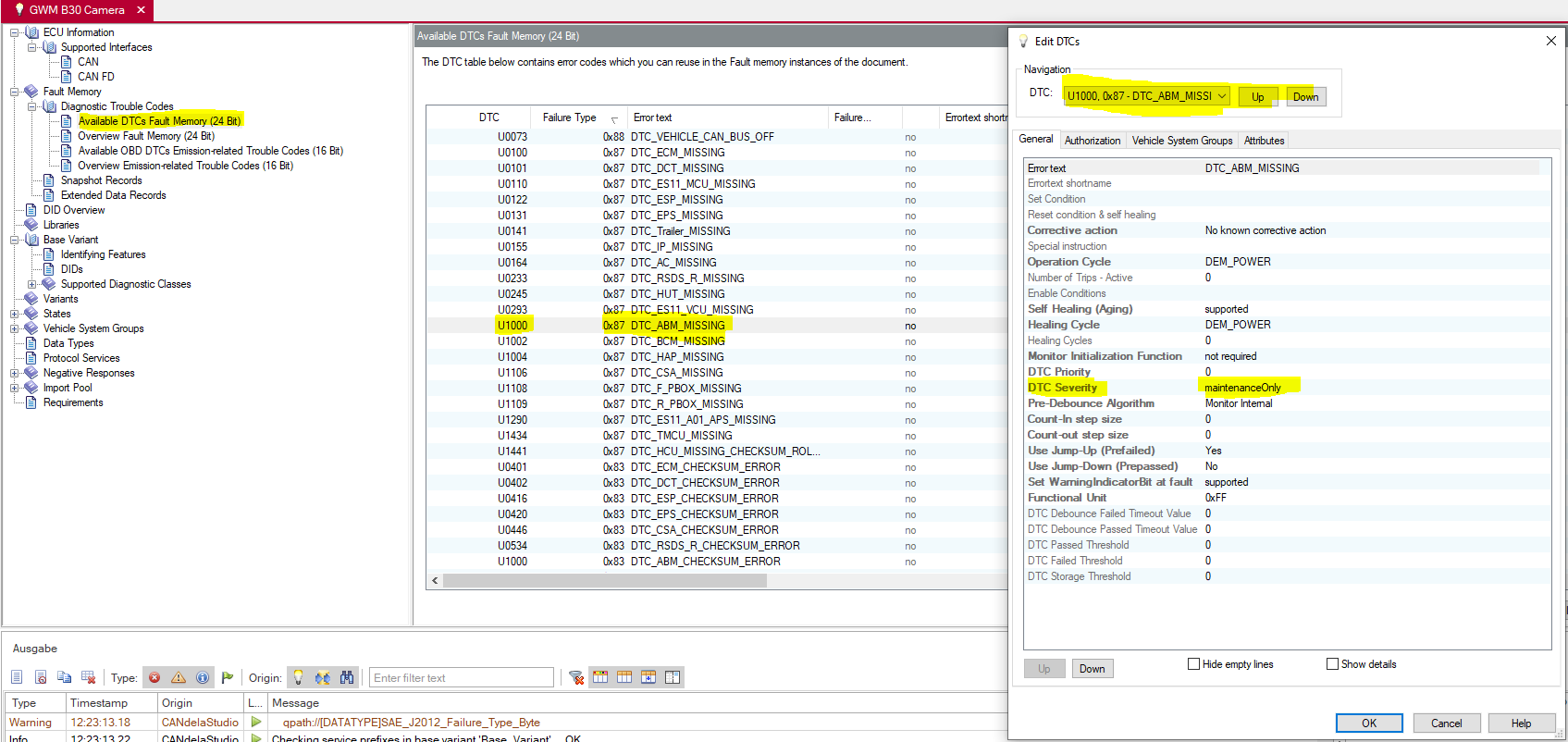




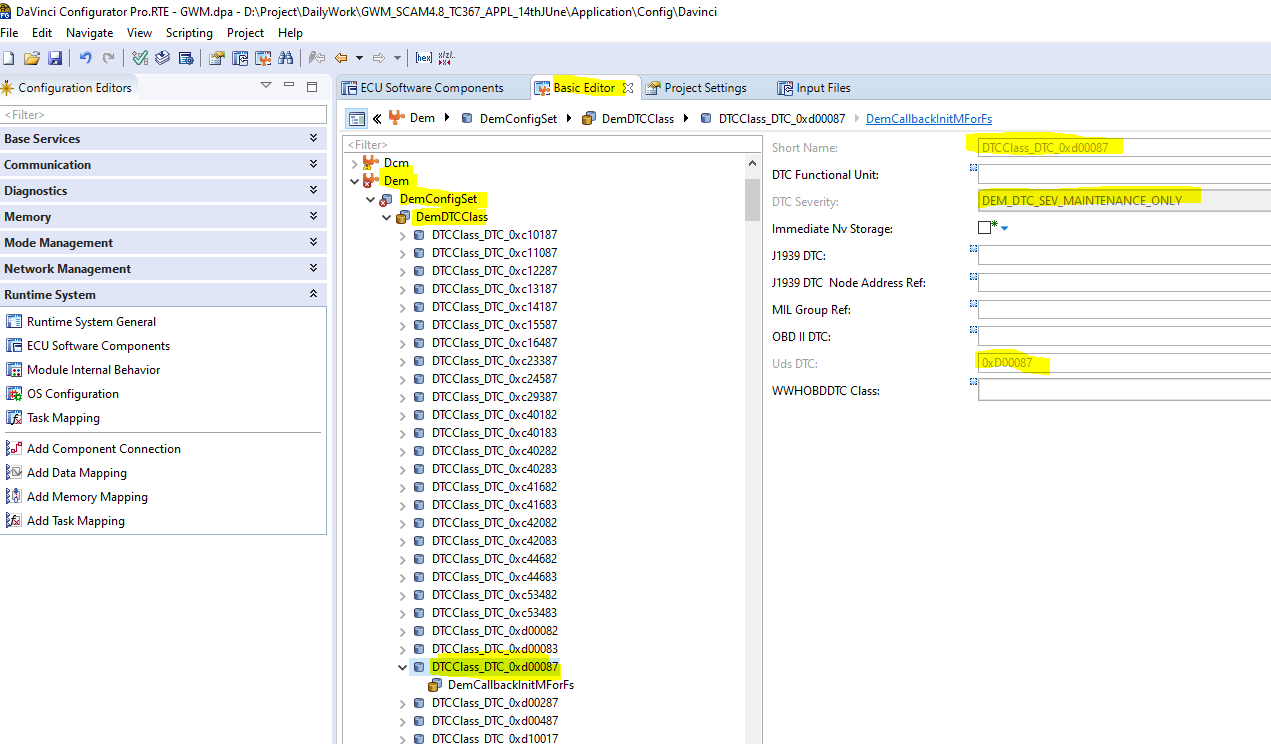
So as per above requirement for DTC U100087 ( D00087) there are two events

1. FAULT\_ABM1\_0x351\_MISSING
2. FAULT\_ABM2\_0x245\_MISSING

This DTC U100087 is mentioned is candela file which we import in Davinci Configurator

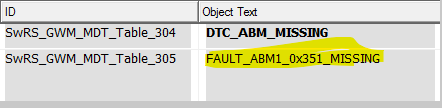


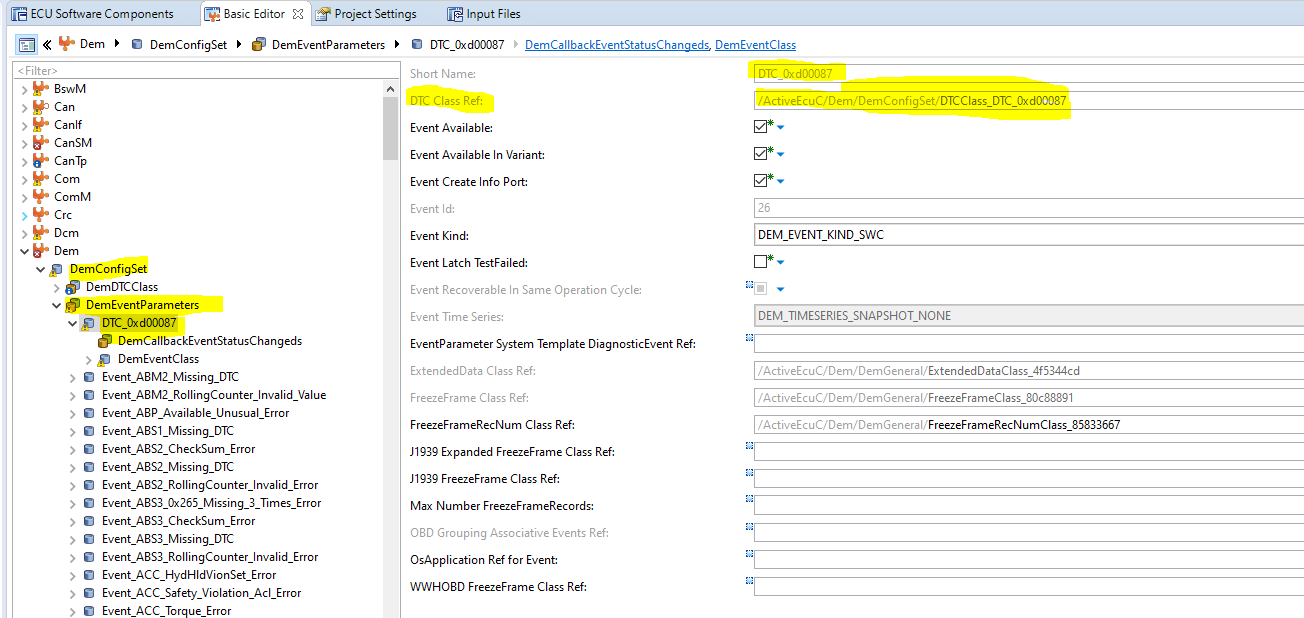
When Candela file import to Configurator below configuration will come from Candela file

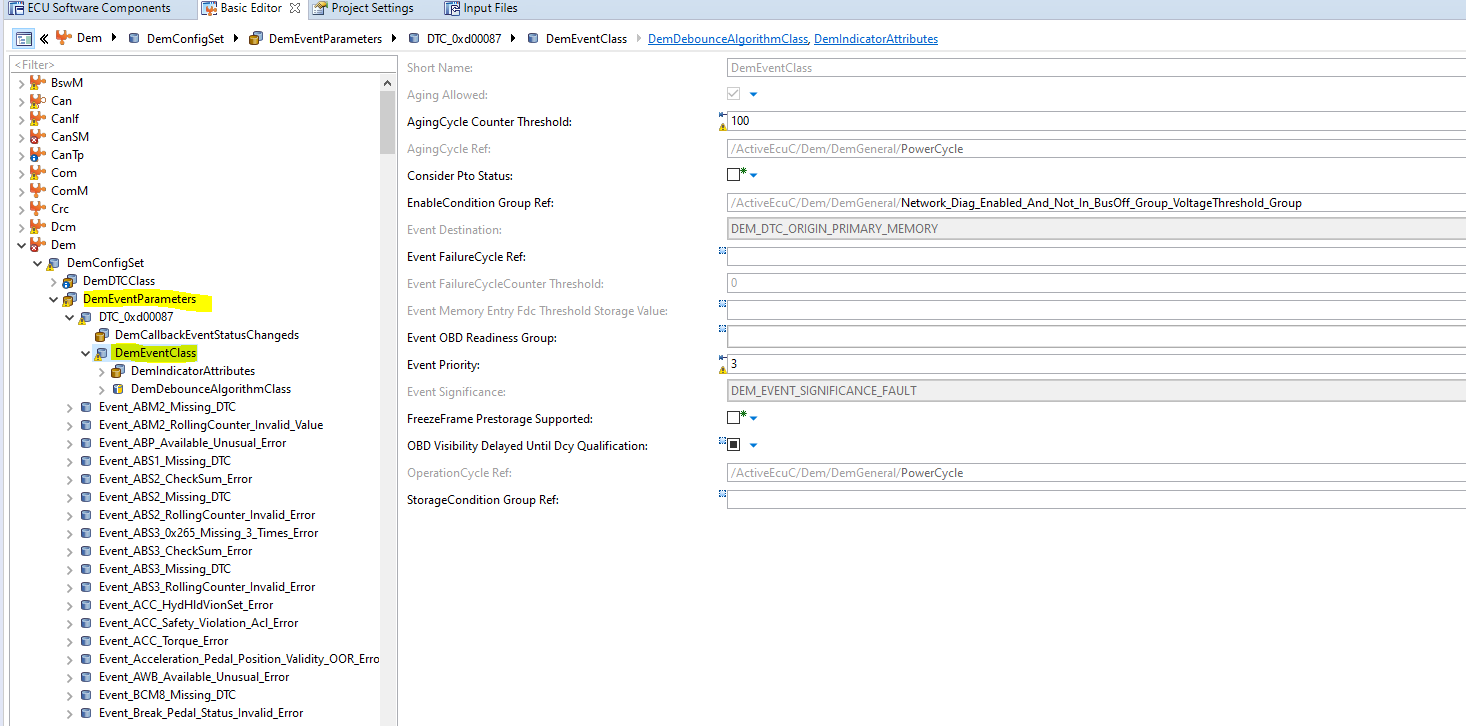


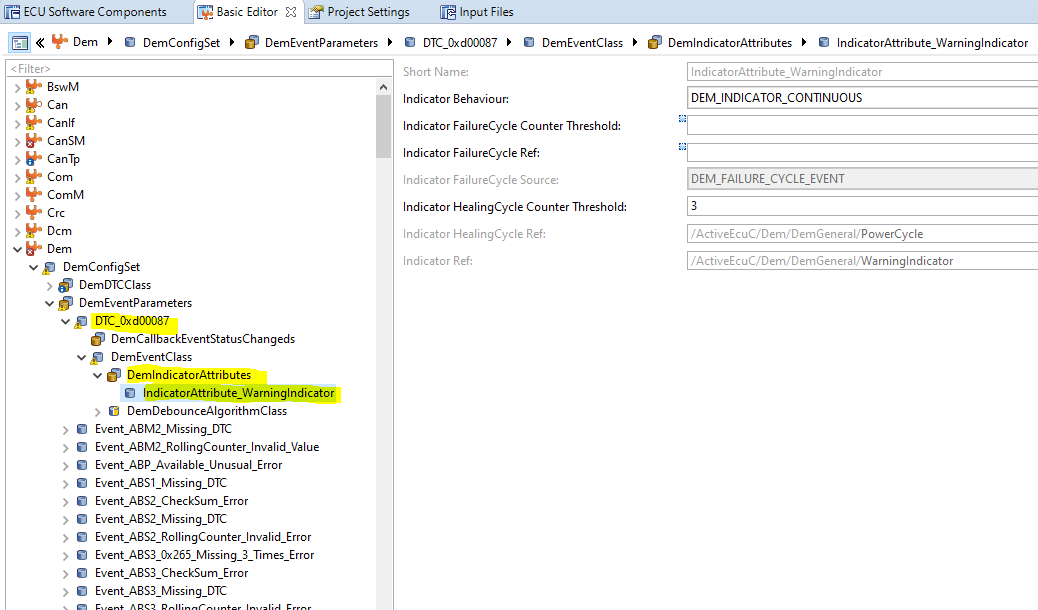
For this DTC there are two events need to configure

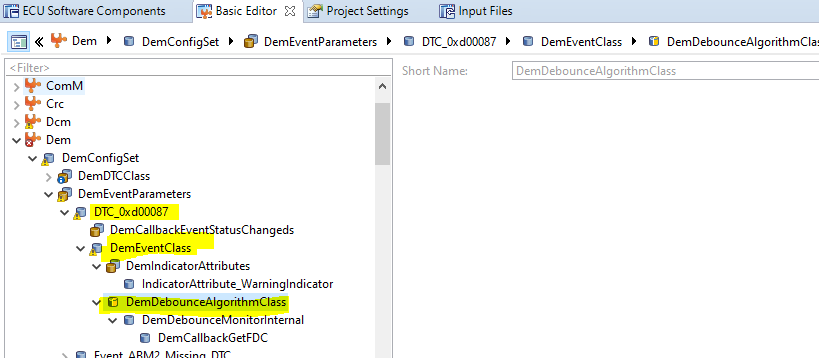
One event(DTC\_0xd00087) is already created default by configurator, which is shown as below inside DemEventParameter this is consider for event (FAULT\_ABM1\_0x351\_MISSING)

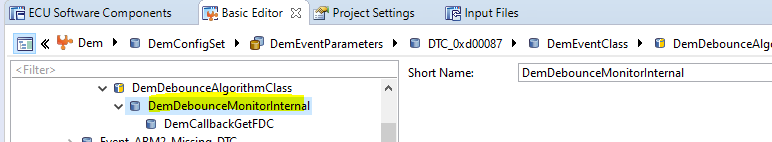


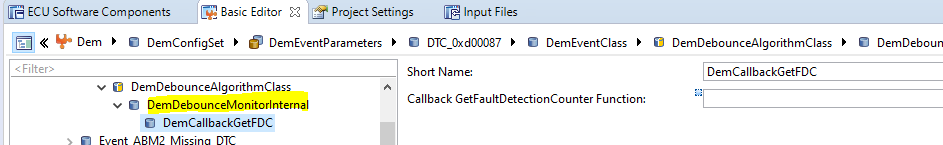




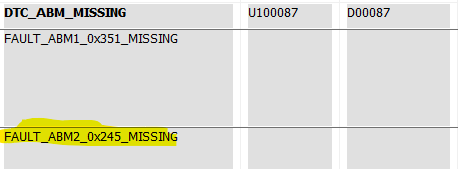


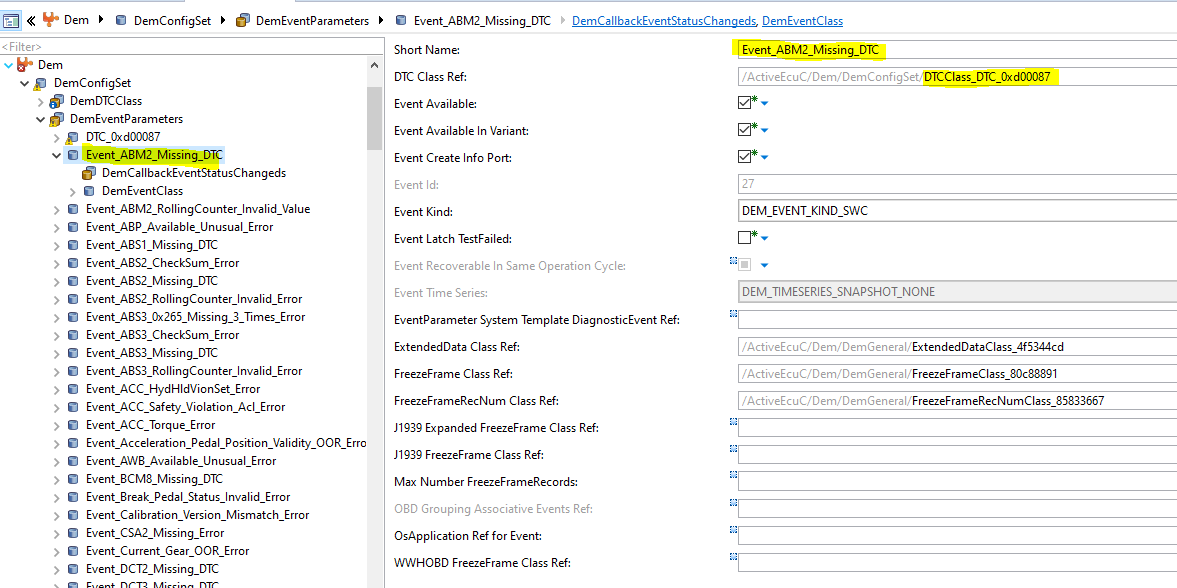


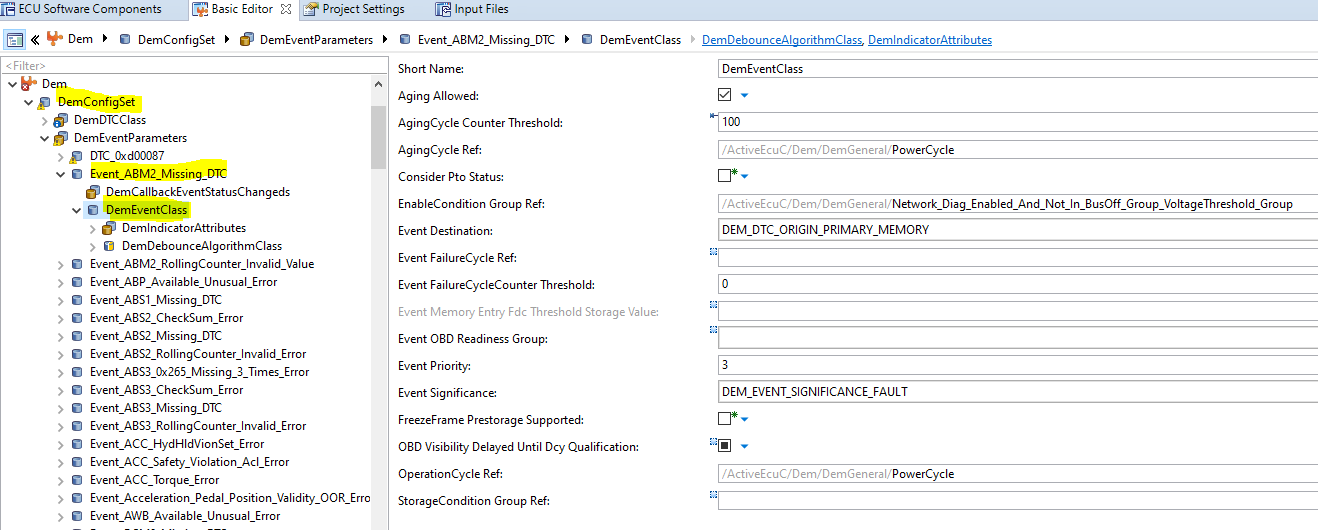


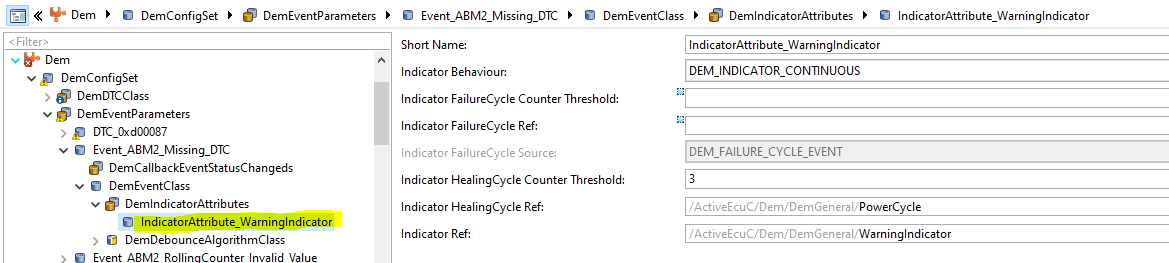


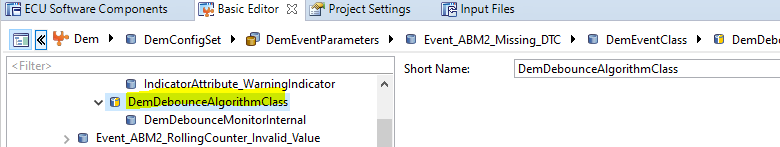
For second event we need to add in DemEventParameters

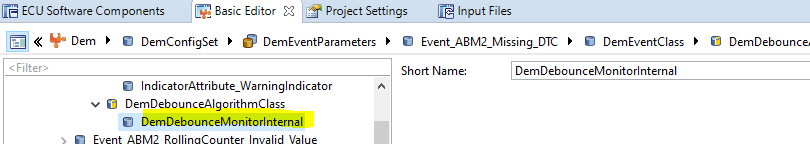






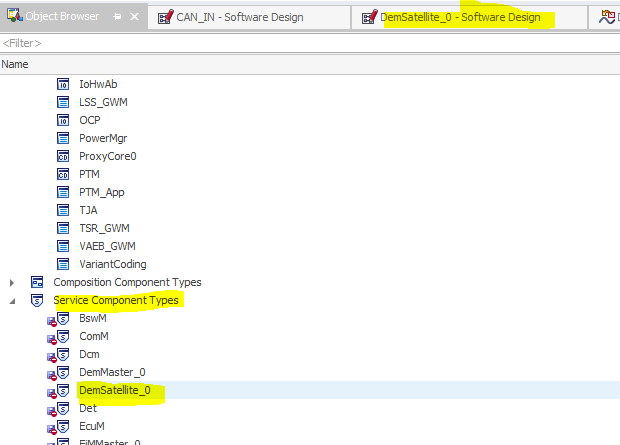






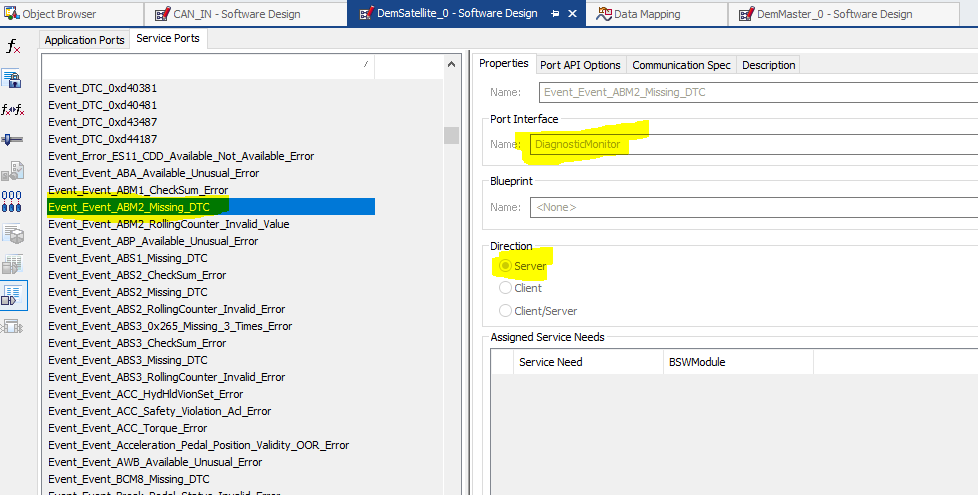
Whatever events comes under DemEventParamter that all comes under service component DemSatellite\_0 ,check below image for understanding

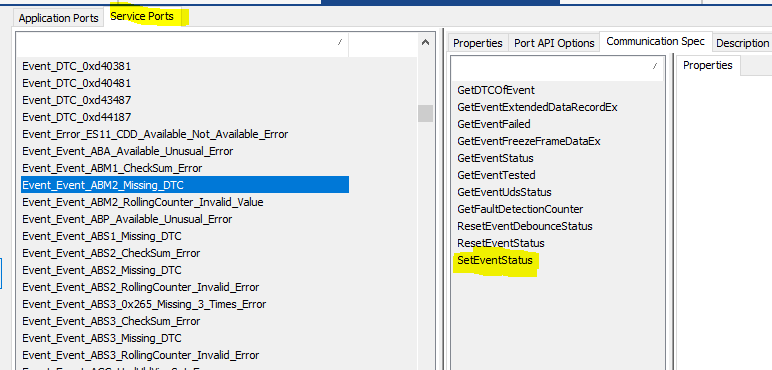




DemSatellite service component

Inside DemSatellite service component Ports are created with name Event\_DemEventParameters , ex:- Event\_Event\_ABM2\_Missing\_DTC and this is server port.





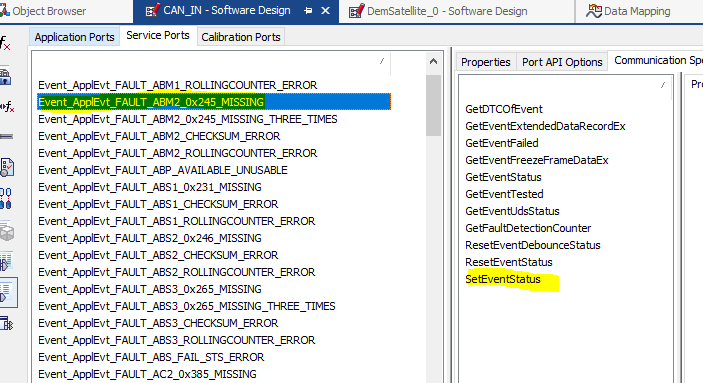
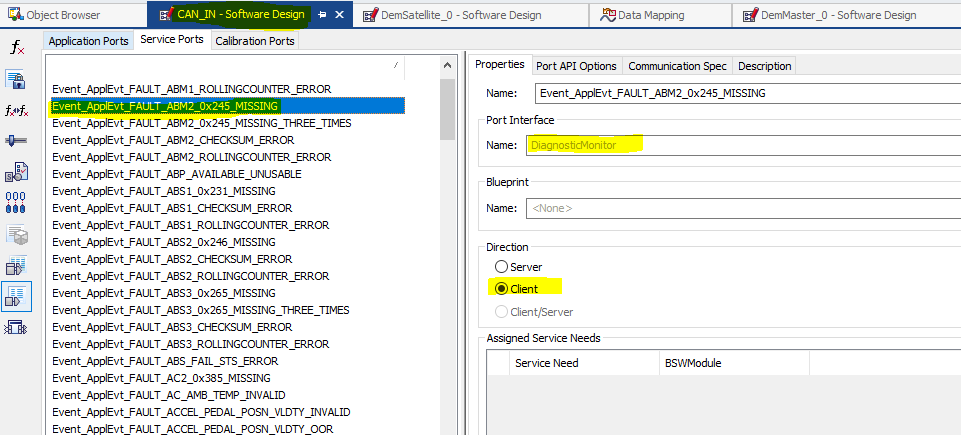
So any application component want to set DTC because of failure of the message then application component need to connect DemSatellite SW component and need to call SetEventStatus function with parameter as fail.

In GWM project there are two component which will set the missing message related event

1. CAN\_IN
2. ADAS\_CAN\_IN --> if applicable

So, check below CAN\_IN module ,it act as client .

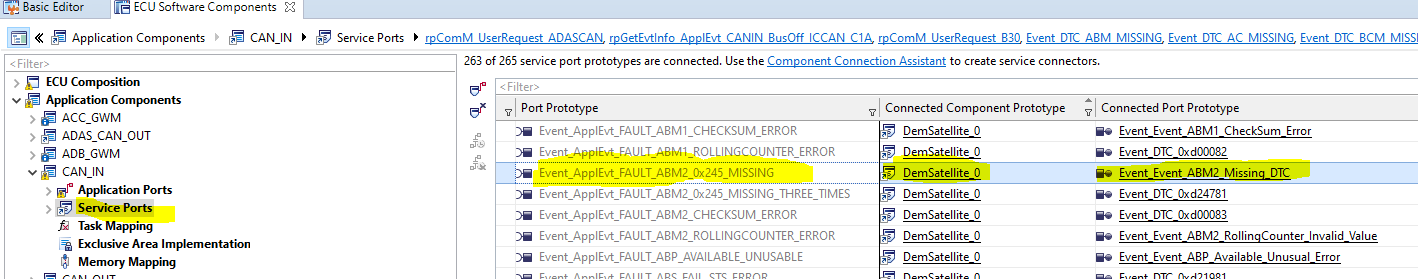
For missing message (0x245)

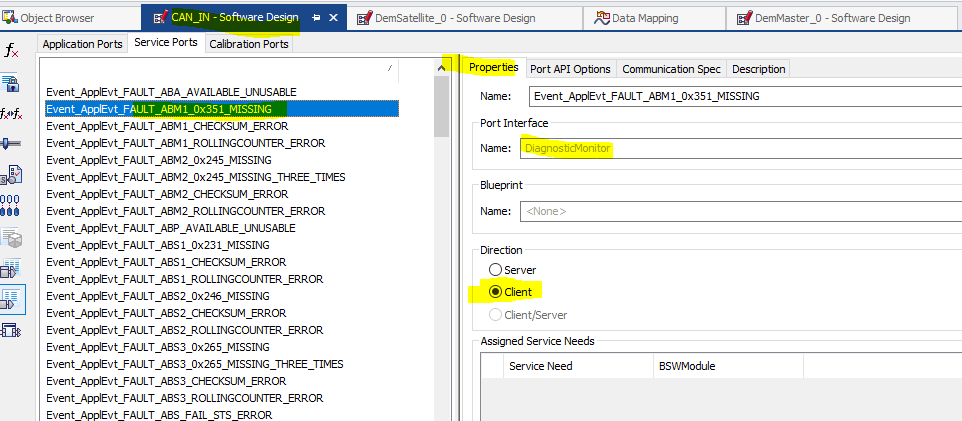
In CAN\_IN software module it will track the message timeout if timeout happens then it will send SetEventStatus as fail otherwise Pass

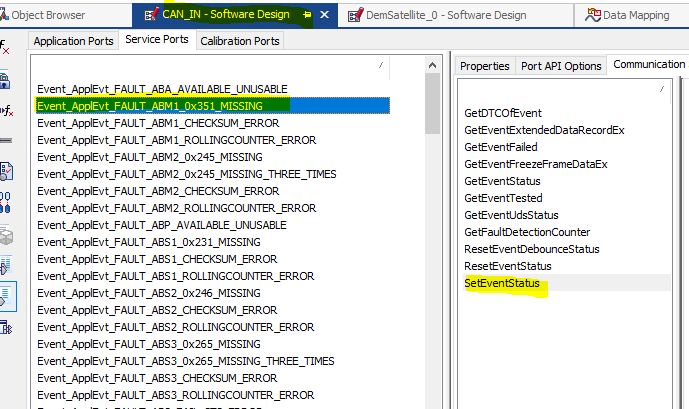
Connection

DemSatellite\_0 ---> CAN\_IN



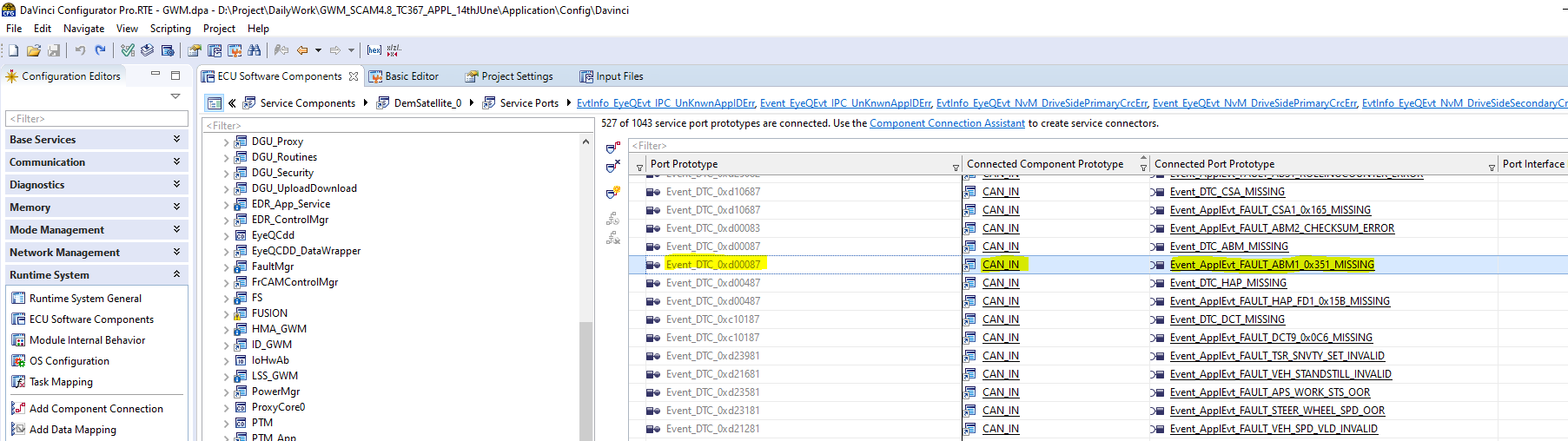
For missing message (0x351)





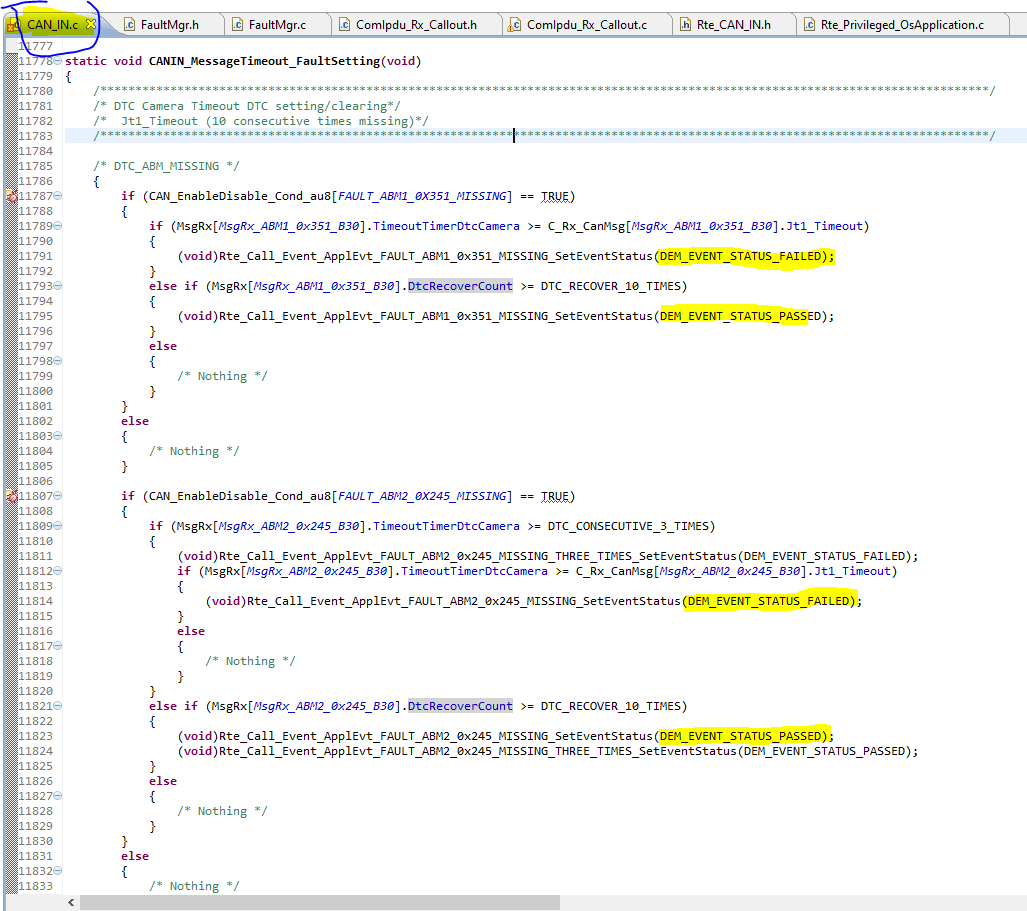
Connection

DemSatellite\_0 ---> CAN\_IN



**Chap 2**

Now, we will check how set event missing message (0x245,0x351), implemented in CAN\_IN software module

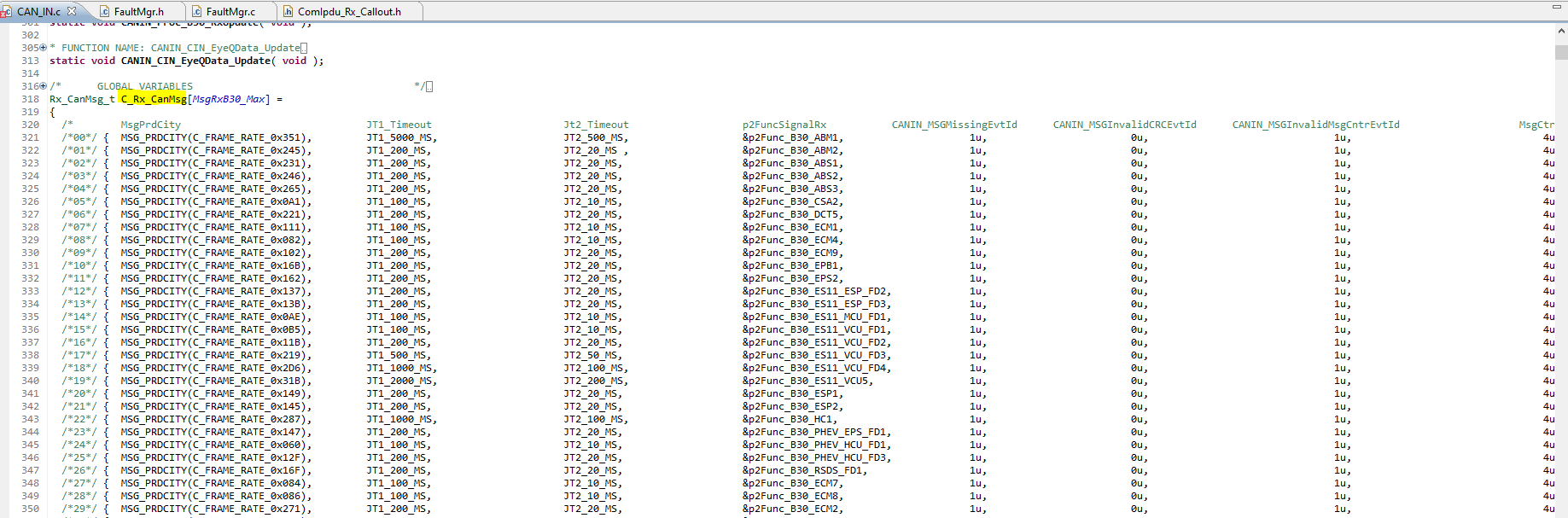


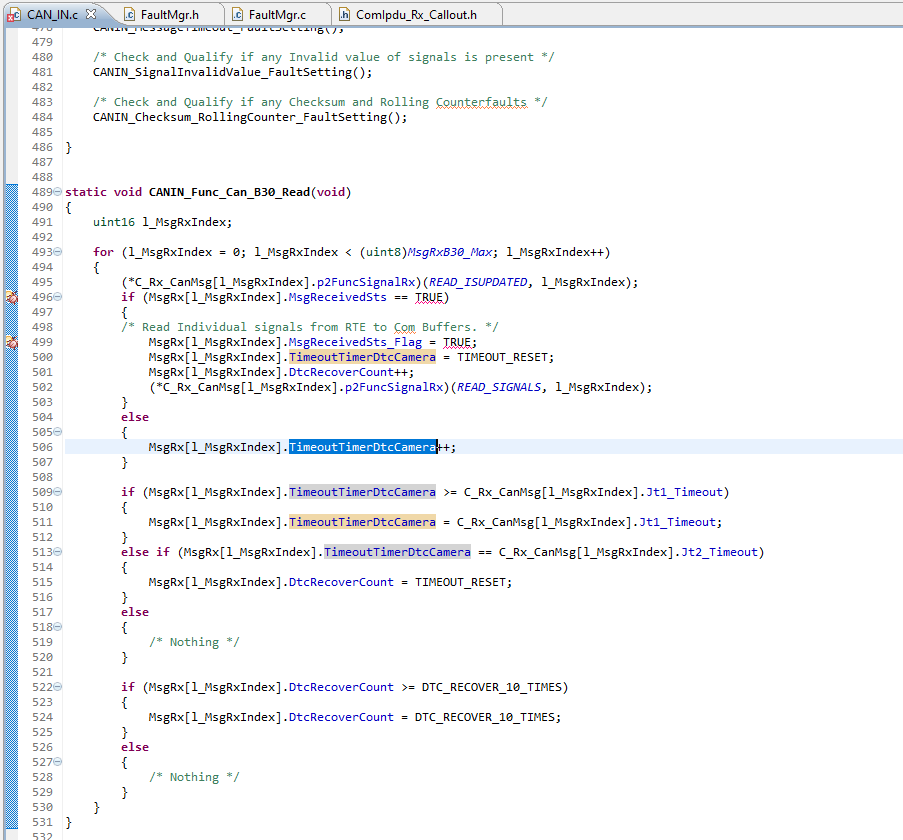
In above code

If (line 11789,11812)--> checking timeout counter reached up to defined count ,if yes then event set as failed

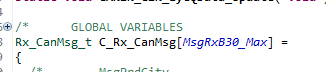
else if (11793,11821)--> Checking if there is a timeout fault and now fault is removed then waiting for recovery time

else --> if there is no timeout fault now or previously then it will do nothing.





So using above code it is understand Timeout and recovery time configured in table



When message received it call if case (line 496) otherwise it will start increment counter (line 506)

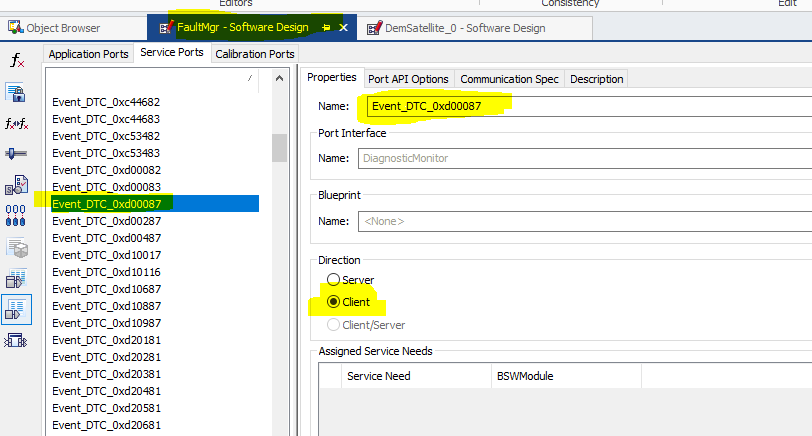
**Chap 3**

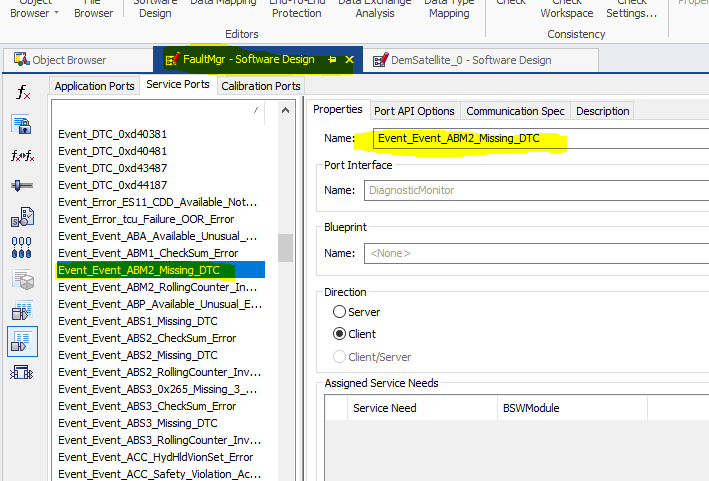
Now, we will check how GetEventUdsStatus / GetEventStatus missing message (0x245,0x351), implemented in FaultManager software module

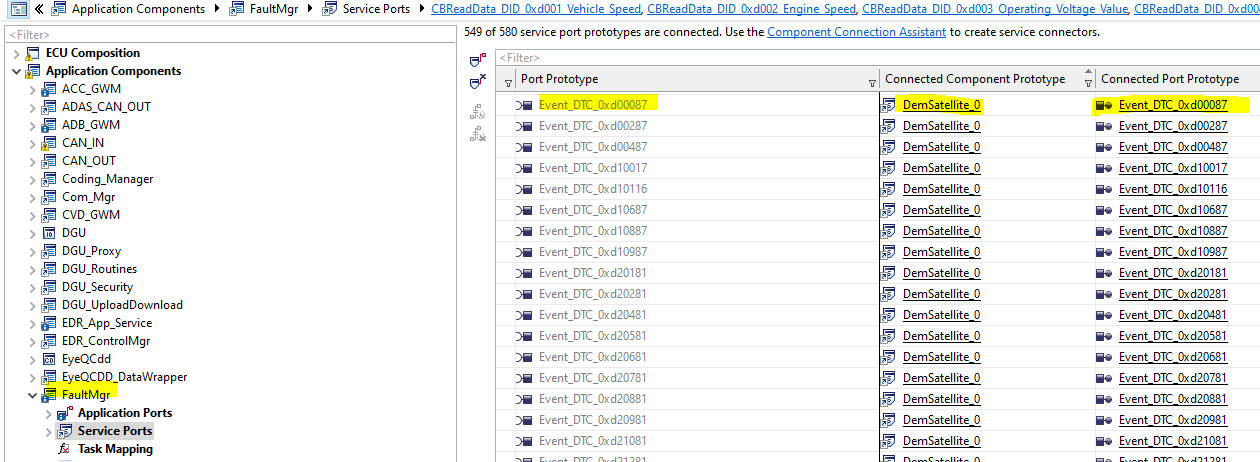
DTC will set using the event status ,So all event status will check in file FaultMgr.c file

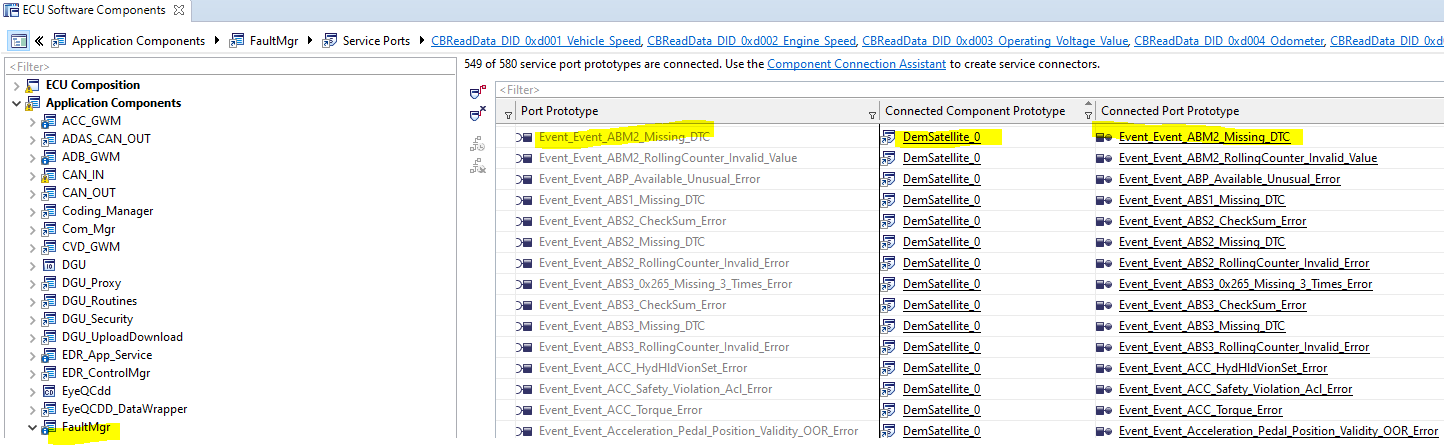
There need to connection between FaultMgr and DemSatellite

Check Below image for the same



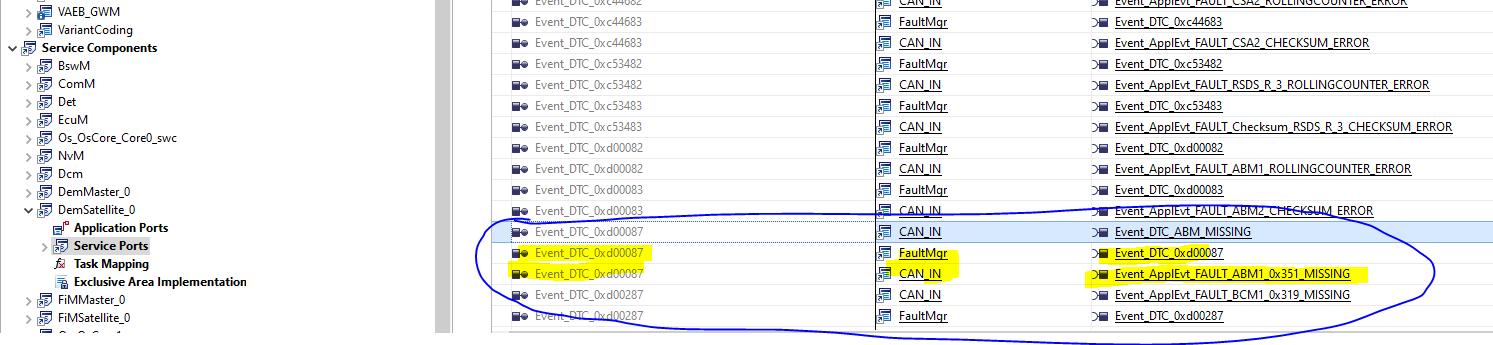


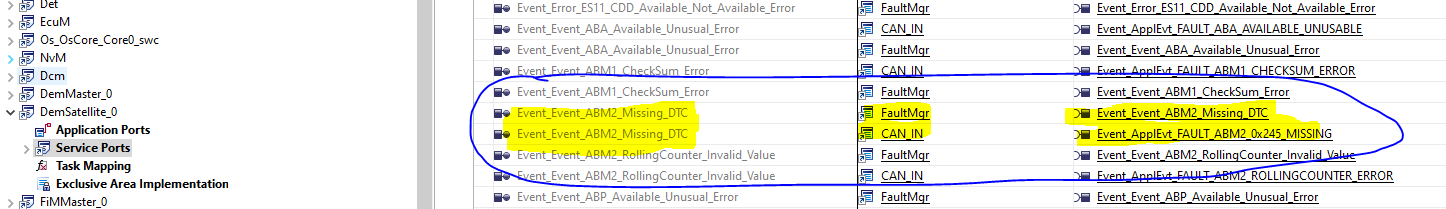




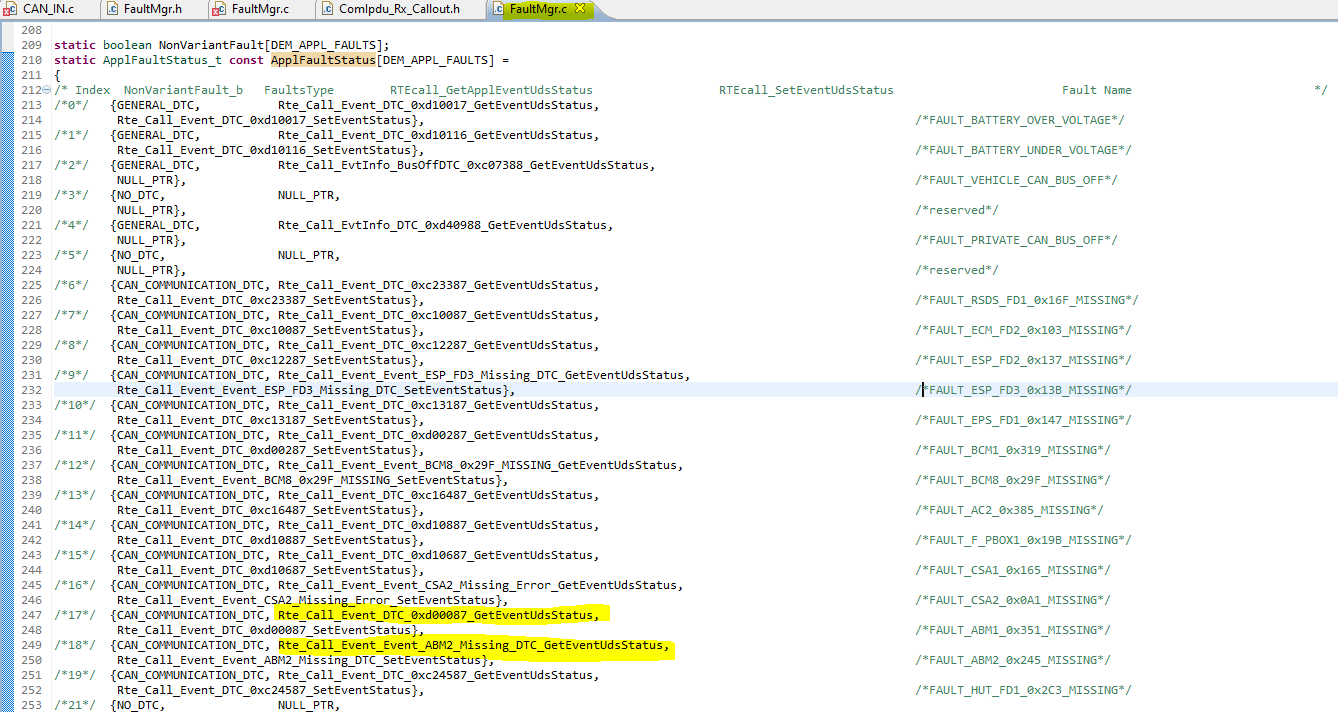
Conclusion is that ,SetEventStatus will happen in Dem satellite Service component, CAN\_IN module will pass parameter Pass of fail using SetEventStatus call and FaultMgr will check event status using GetEventUdsStatus or GetEventStatus

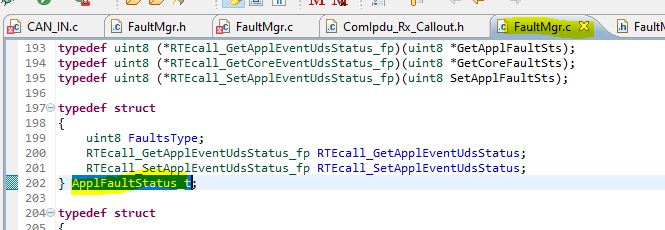
So for this reason two connection for Demstatellite for each event ,Check below image for the same



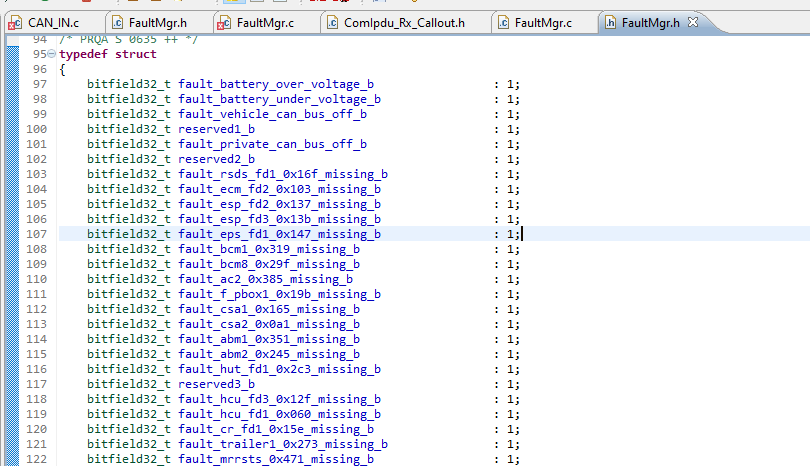


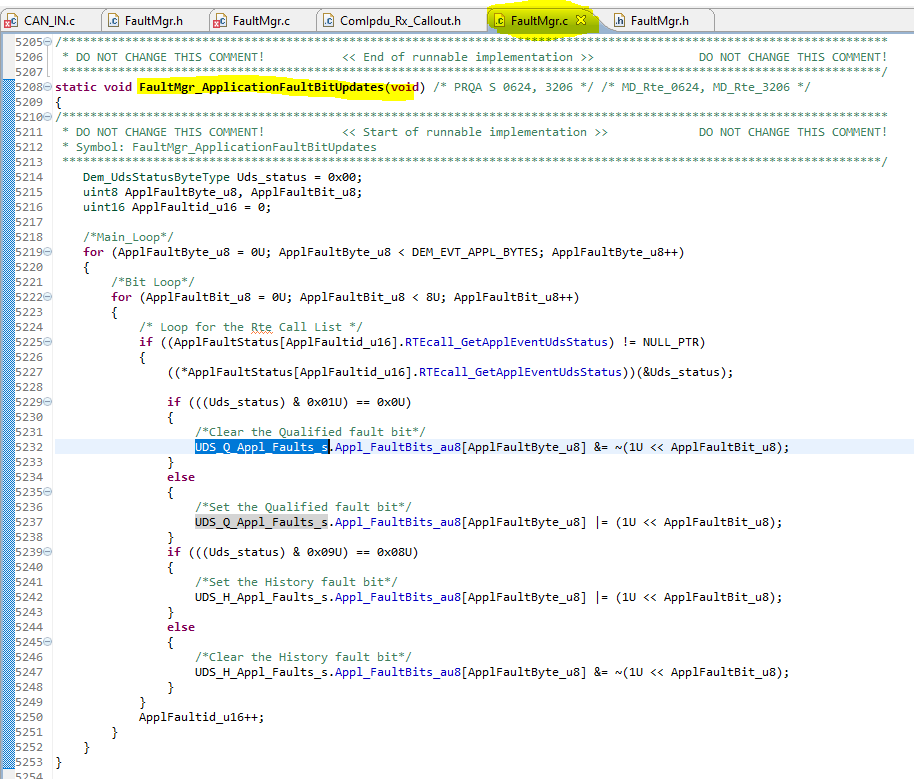
Check below code one table is created for ApplFaultStatus in FaultMgr.c file





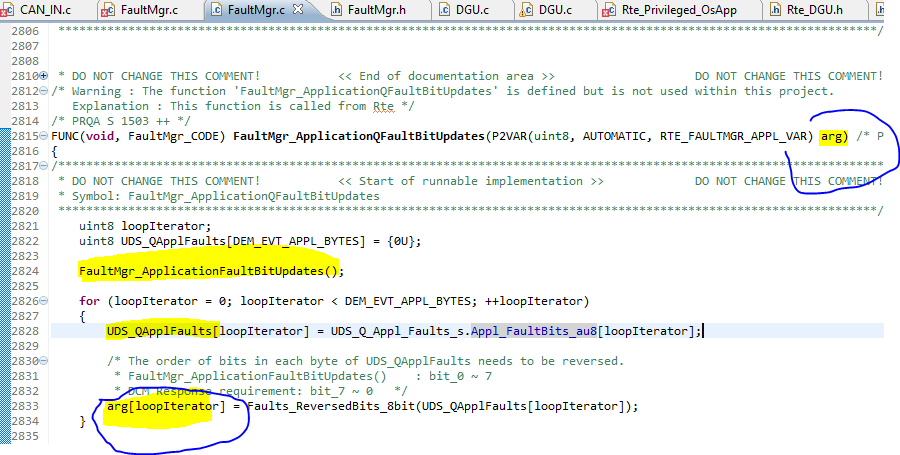
In below function **FaultMgr\_ApplicationFaultBitUpdates(void)**,each event status is checking and updating the structure UDS\_Q\_Appl\_Faults\_s for Qualified fault and UDS\_H\_Appl\_Faults\_s for History fault , This structure is nothing but for each fault one bit is giving and they are updating that one ,is that fault active or not





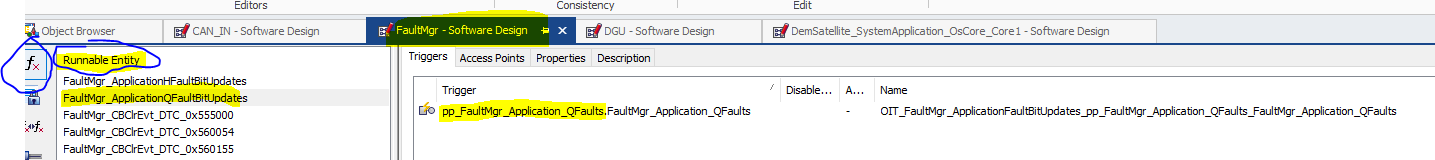
To call above function below function “**FaultMgr\_ApplicationQFaultBitUpdates**” used

And in this function arg updated with Qualified fault status

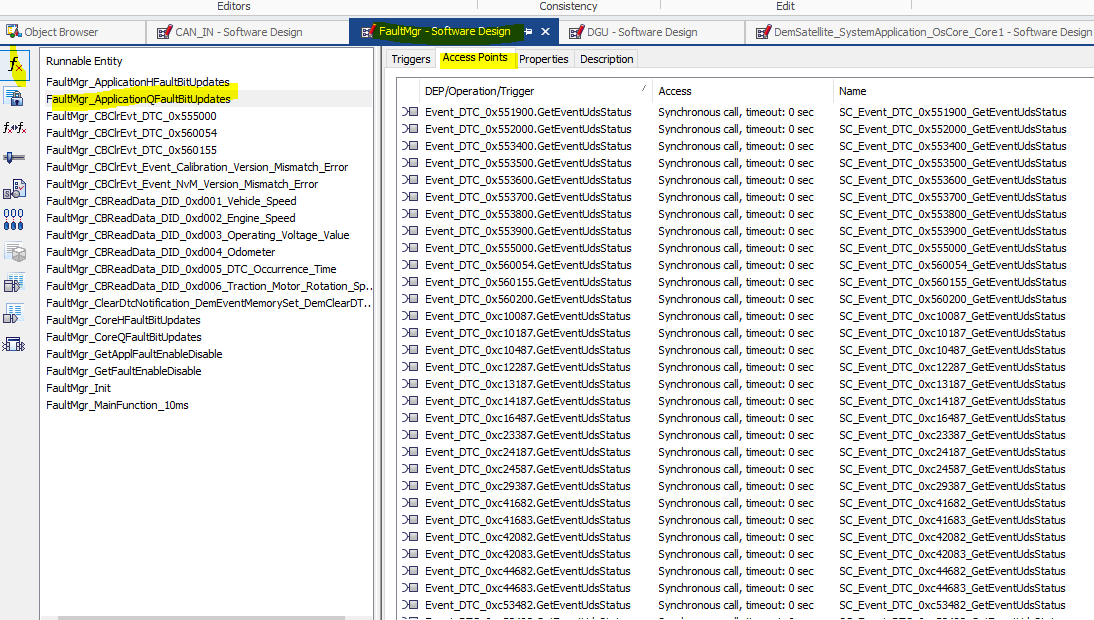


Now look how above function/Runnable “**FaultMgr\_ApplicationQFaultBitUpdates** “ mapped toDGU SW component

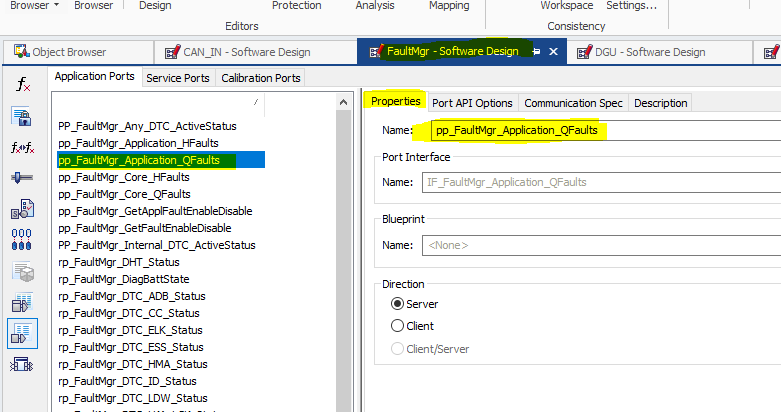
The Function “**FaultMgr\_ApplicationQFaultBitUpdates** “ will trigger by using port pp\_FaultMgr\_Application\_QFaults and this port act as server ,check below image for the same

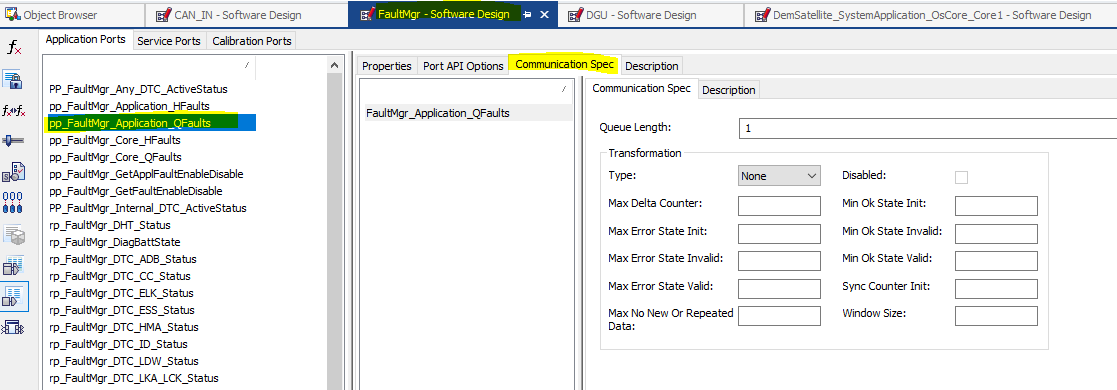


And this function have below access Points



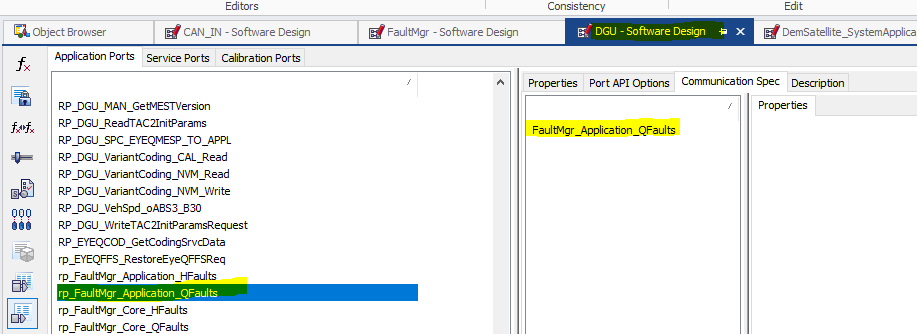
Port pp\_FaultMgr\_Application\_QFaults act as server



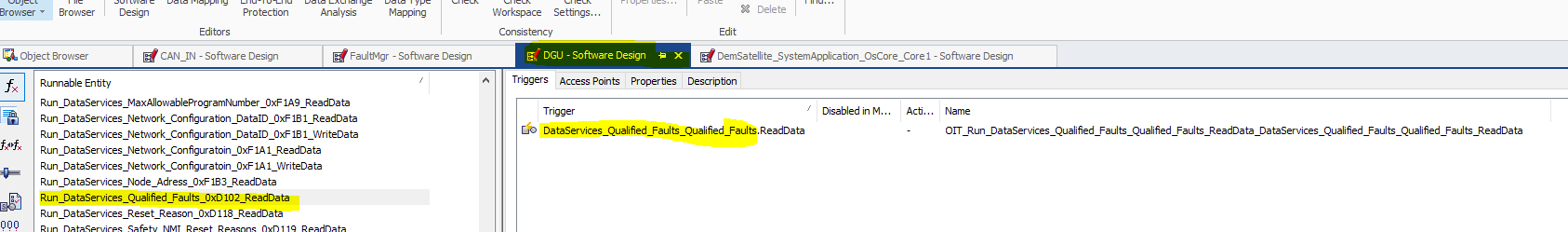


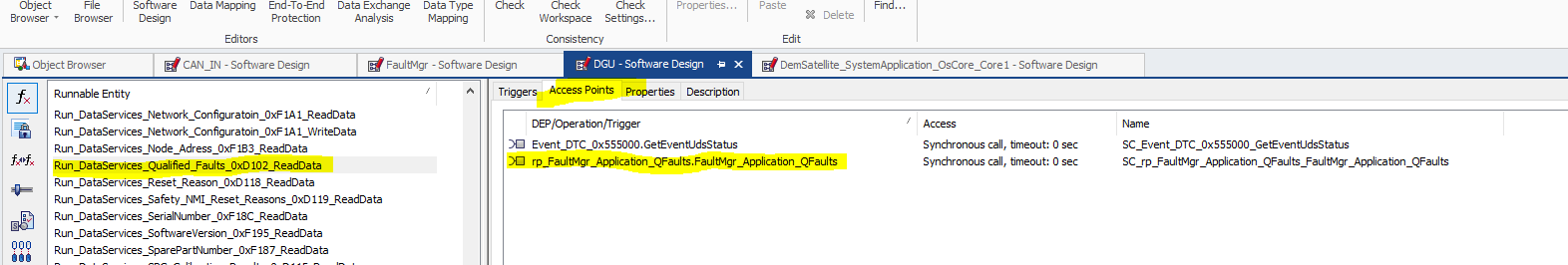
Port rp\_FaultMgr\_Application\_QFaults acts as client



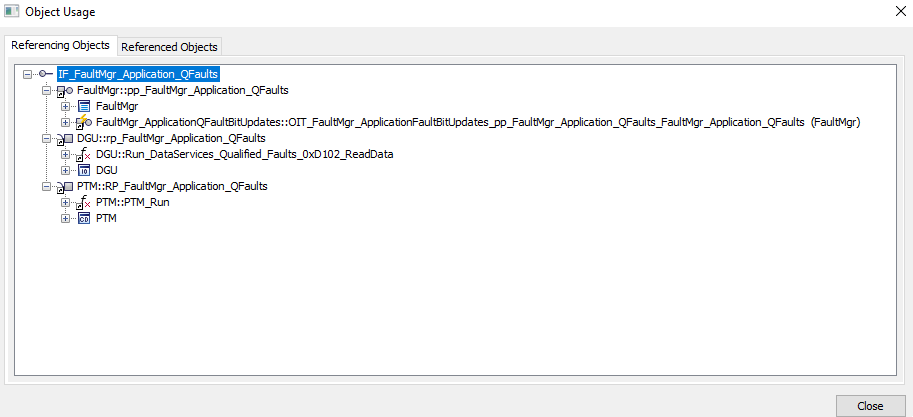


And this port(rp\_FaultMgr\_Core\_QFaults) is mapped to access point of Runnable Run\_DataServices\_Qualified\_Faults\_0xD102\_ReadData ,check image for the same.

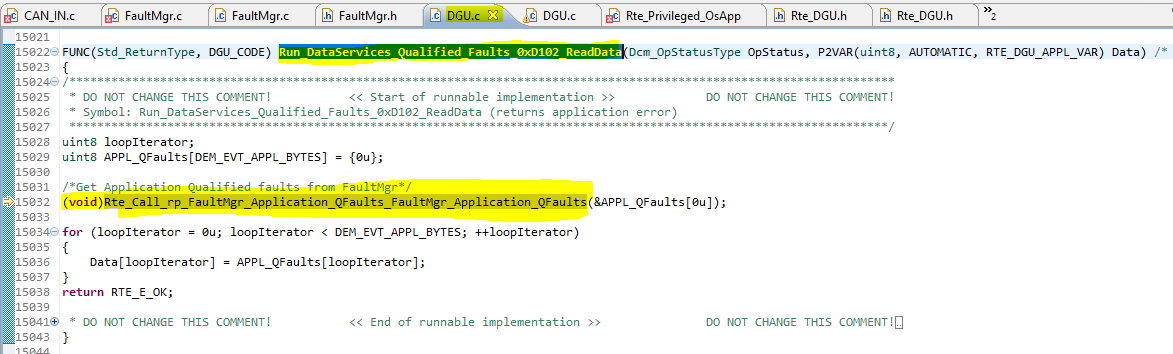




In sumarry ,Below image Object Usage



And Runnable **Run\_DataServices\_Qualified\_Faults\_0xD102\_ReadData** in DGU.c Implementation as below



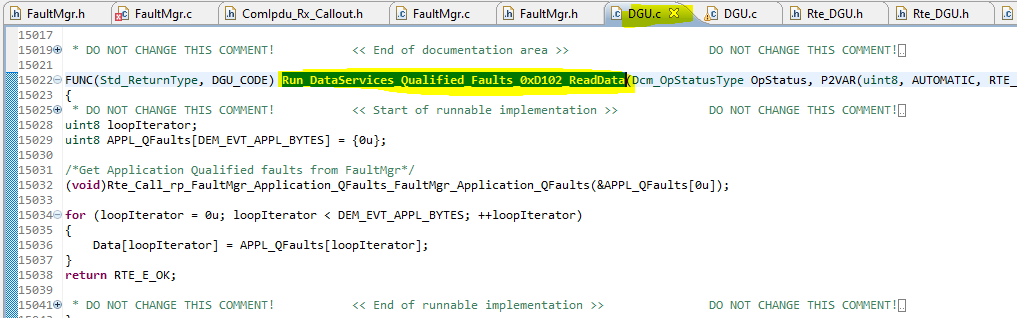
And this RTE call (**void**)Rte\_Call\_rp\_FaultMgr\_Application\_QFaults\_FaultMgr\_Application\_QFaults(&APPL\_QFaults[0u]);

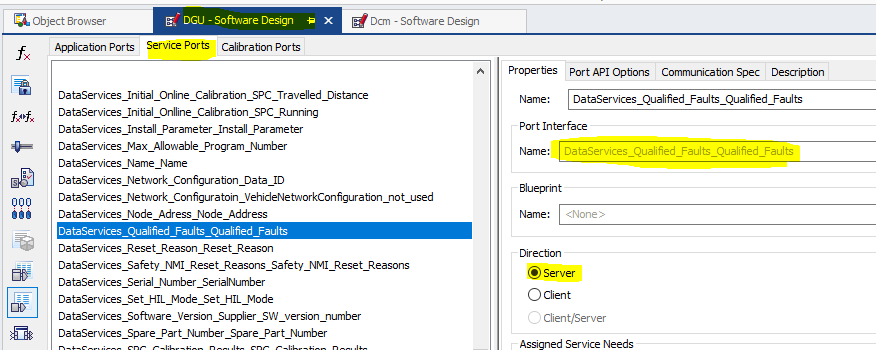
is nothing but call to FaultMgr file which we explained till now

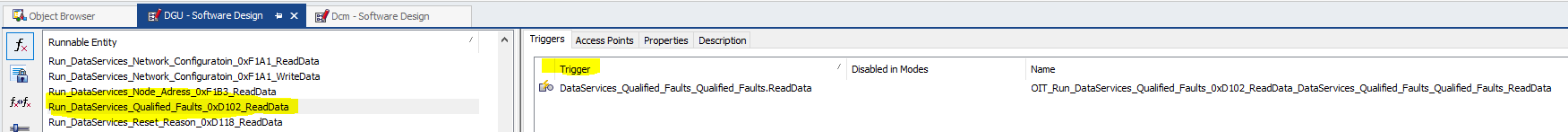
**Chap 4**

The DGU application component is mapped to service component DCM

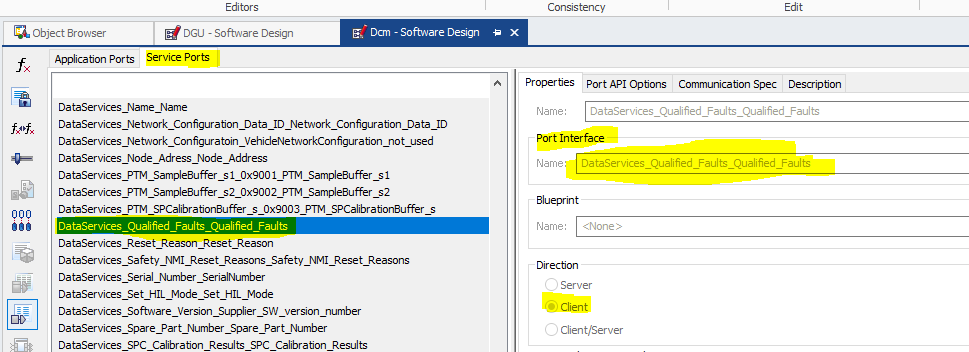
This DCM will trigger on each 10msec

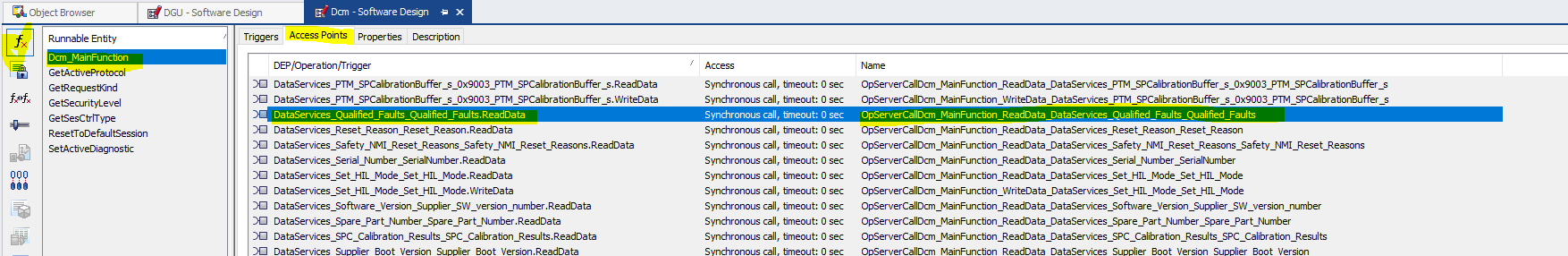


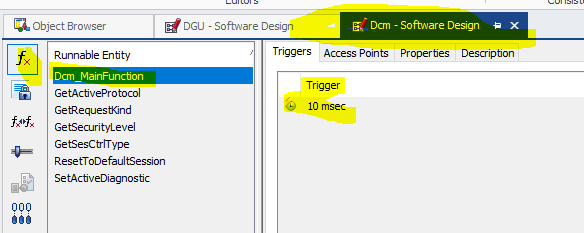




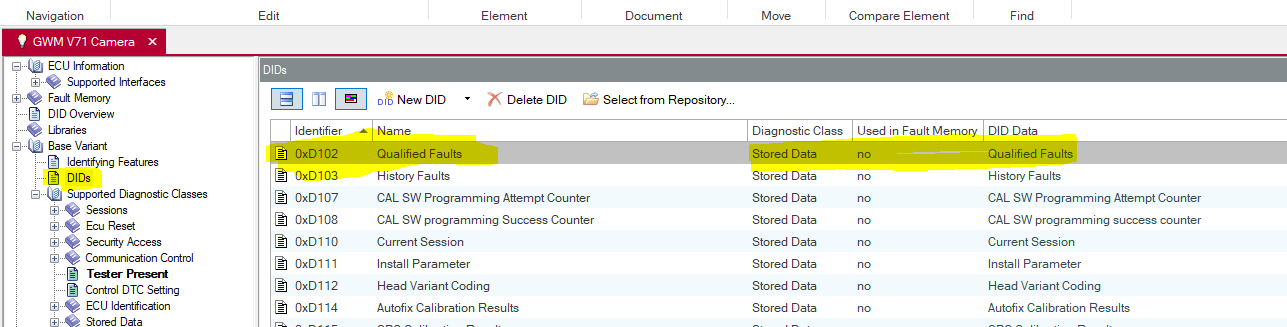
Dcm



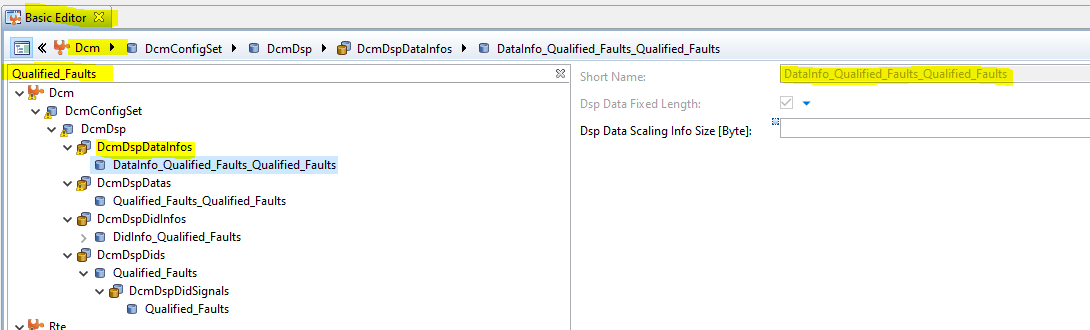


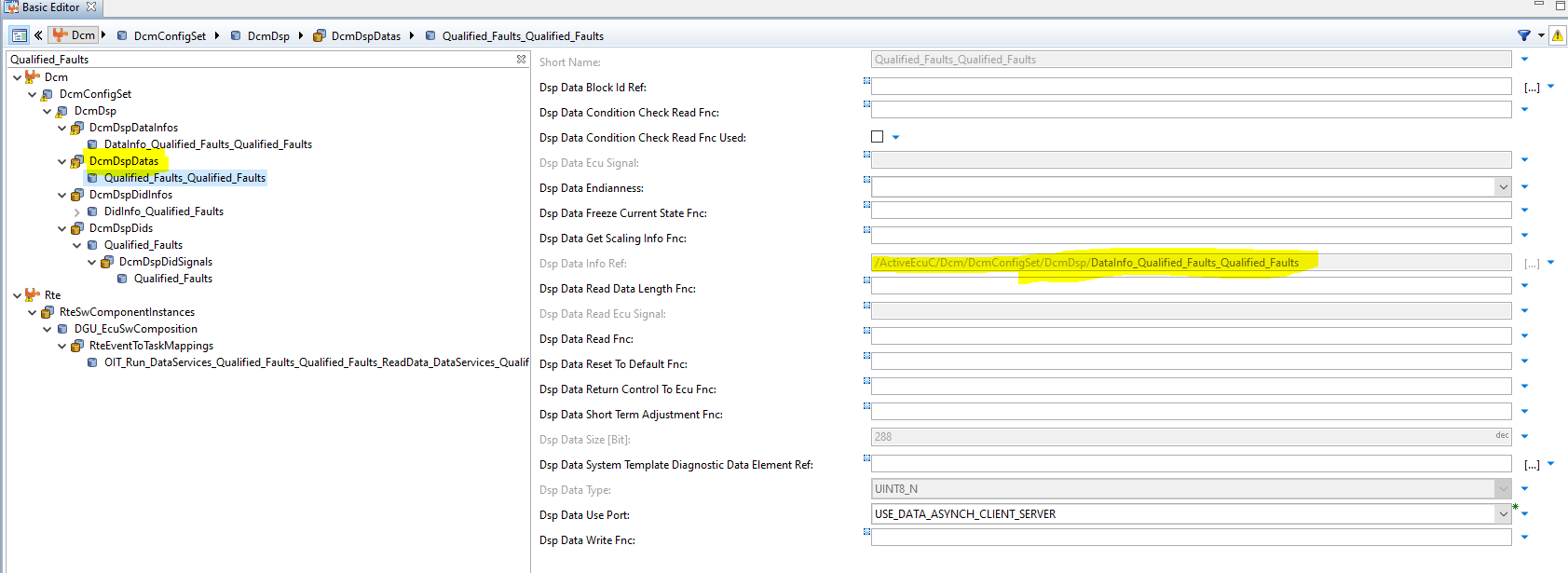


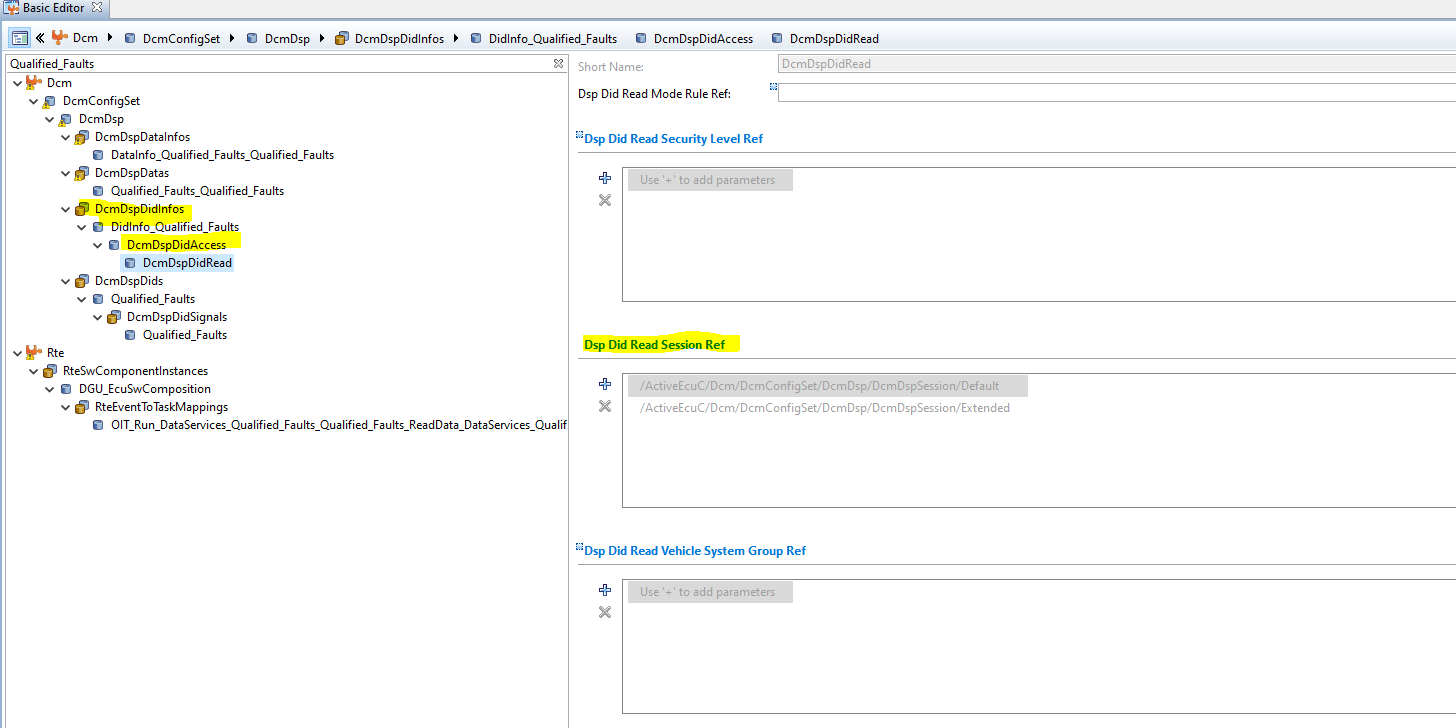
In DCM this service component will add after importing of CDD, Please check below

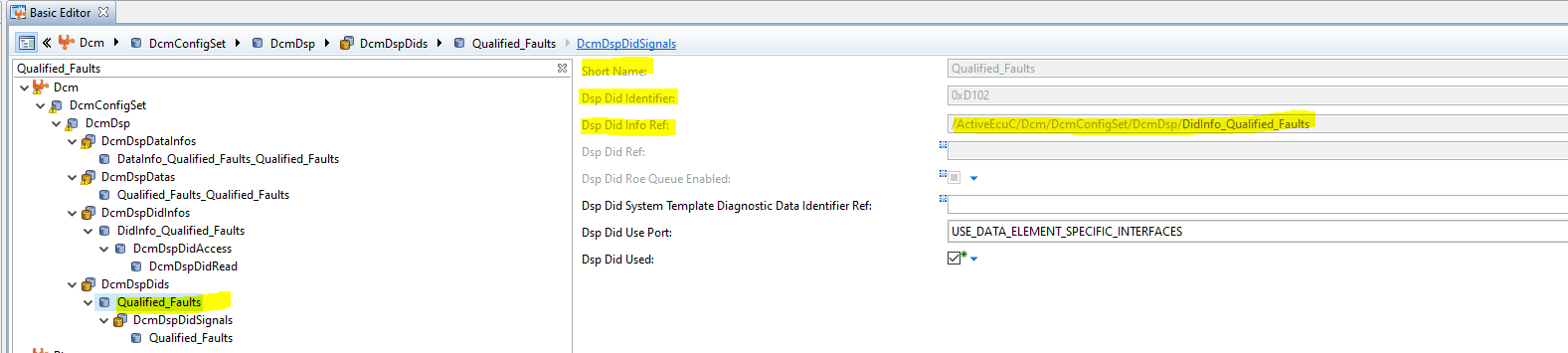


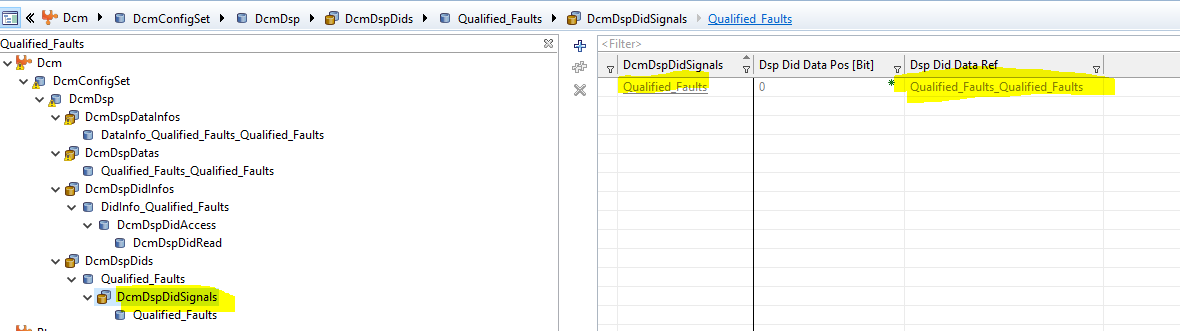
Once above CDD import below thigs will visible

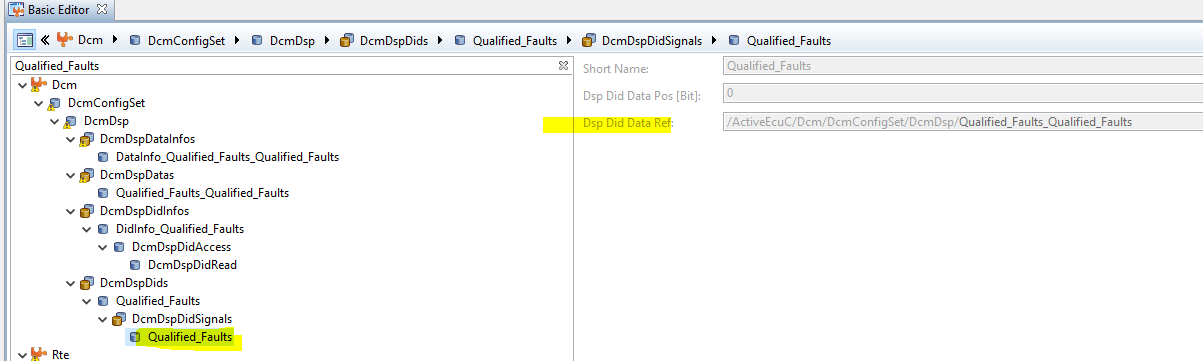


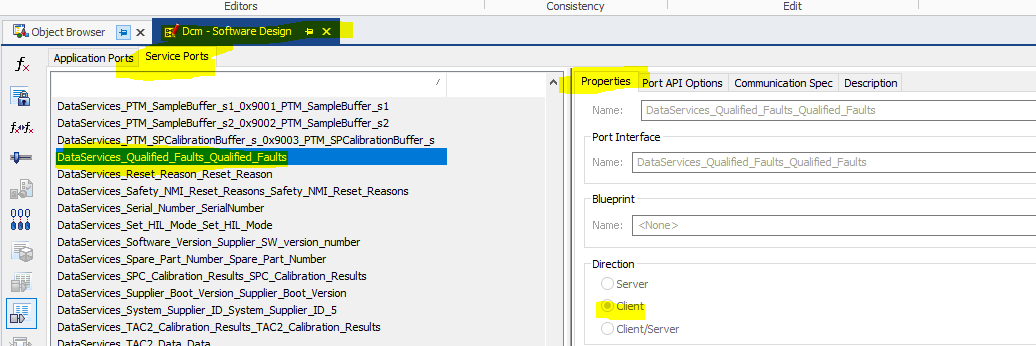










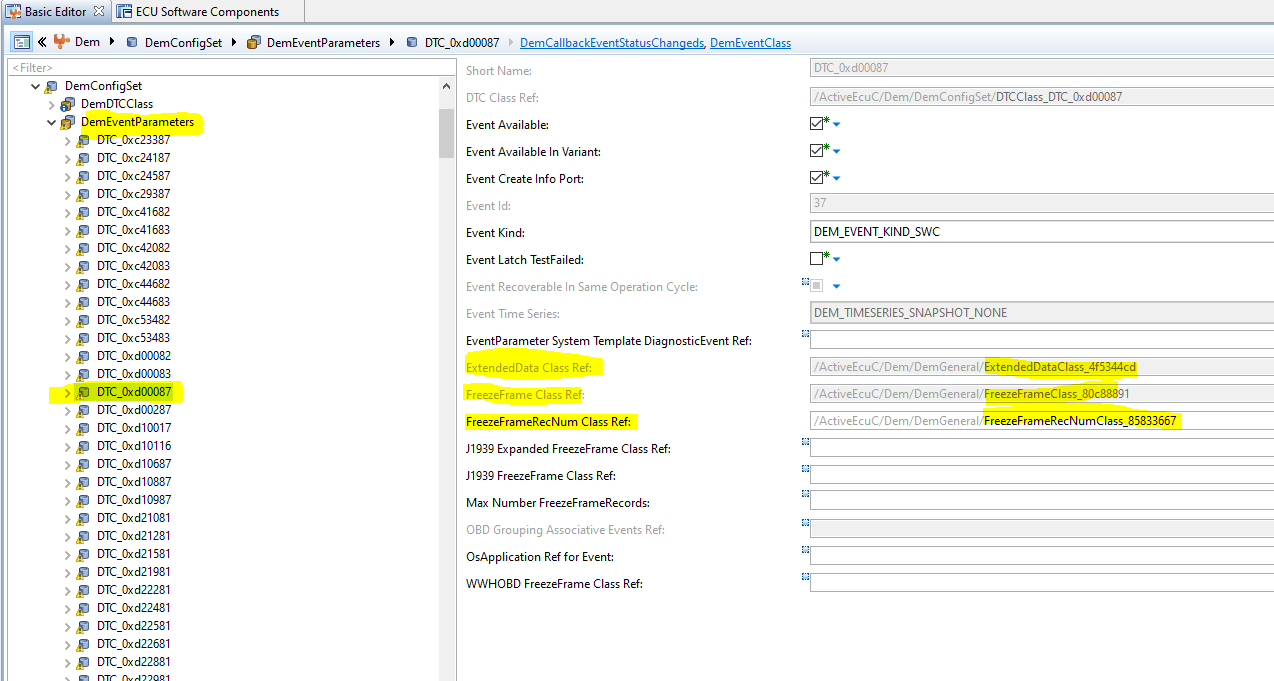




**Chap 5**

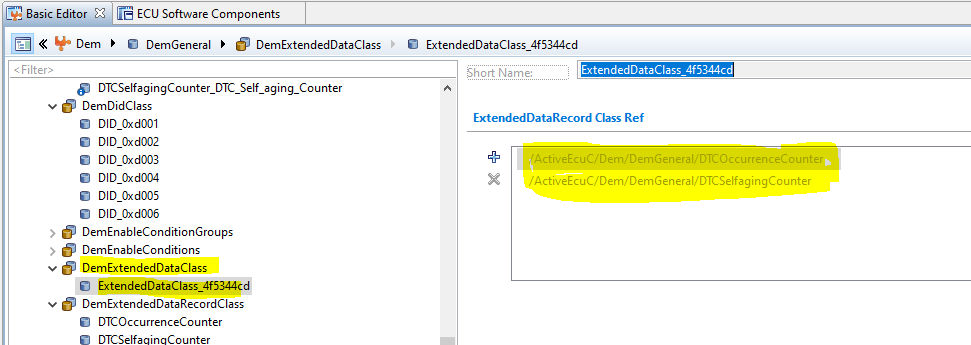
**Extended Data and Freeze Frame Data**

For each even there is Extended Data and Freeze Frame Data which we configured in Event section ,check below event 0xd0087

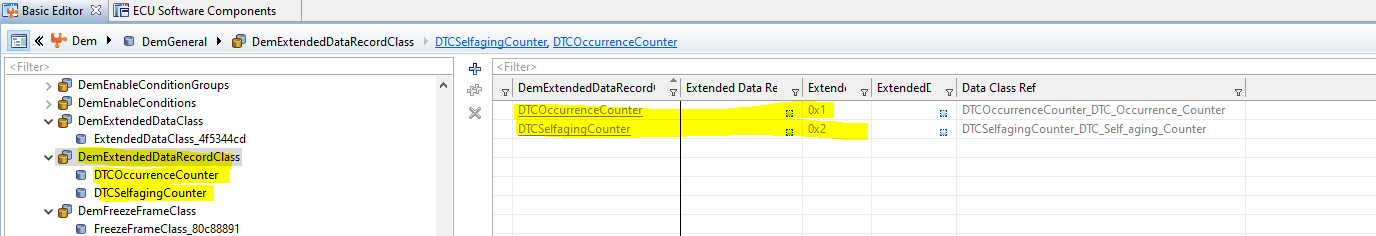


Extended class reference is ExtendedDataClass\_4f5344cd this we manually added.

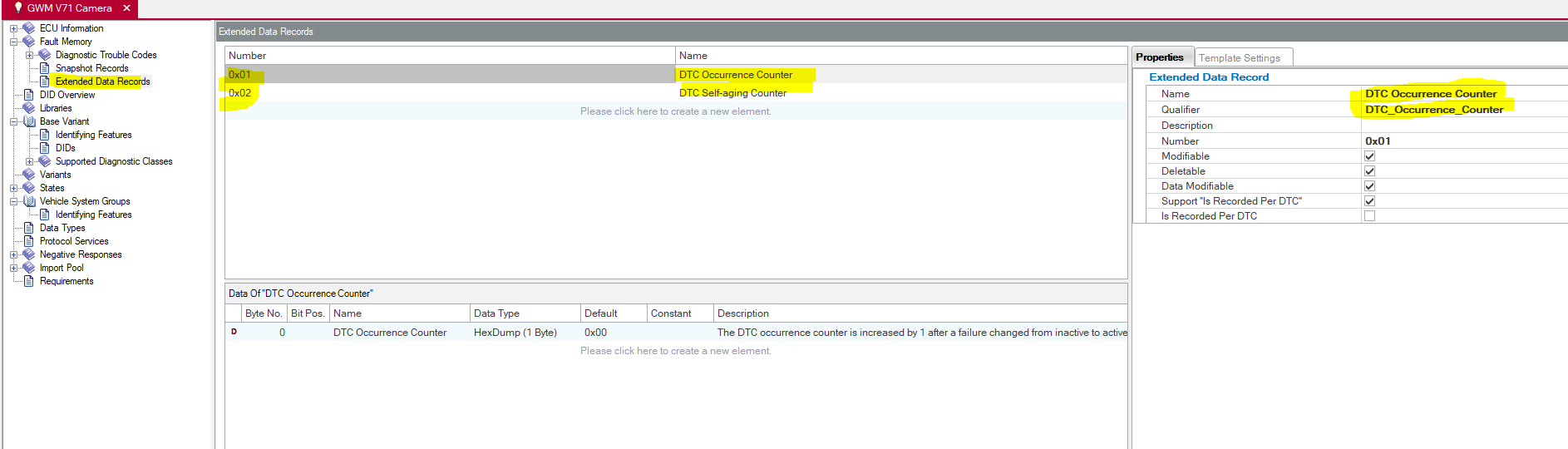
And ExtendedDataRecord Class Ref we configured

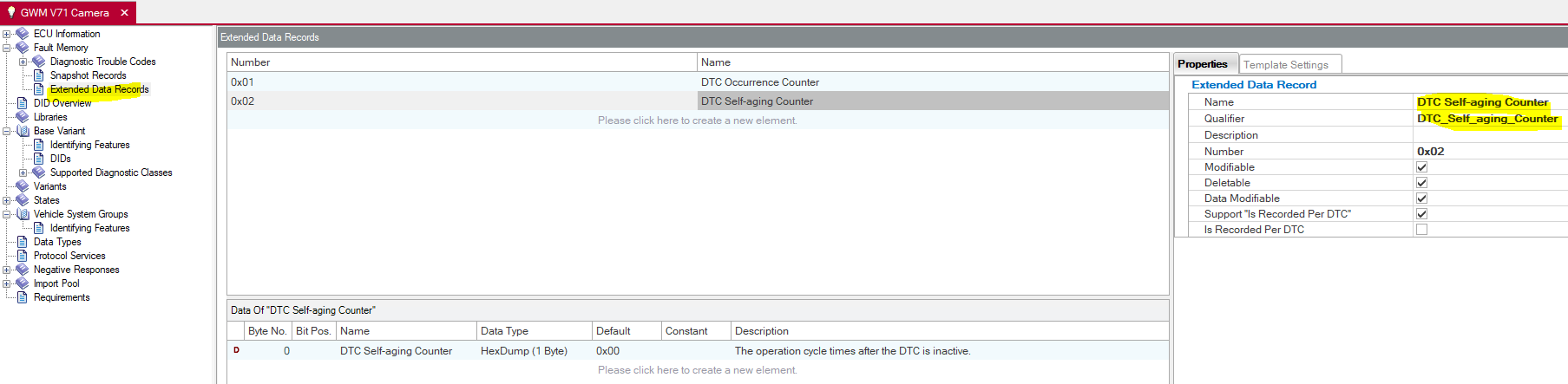


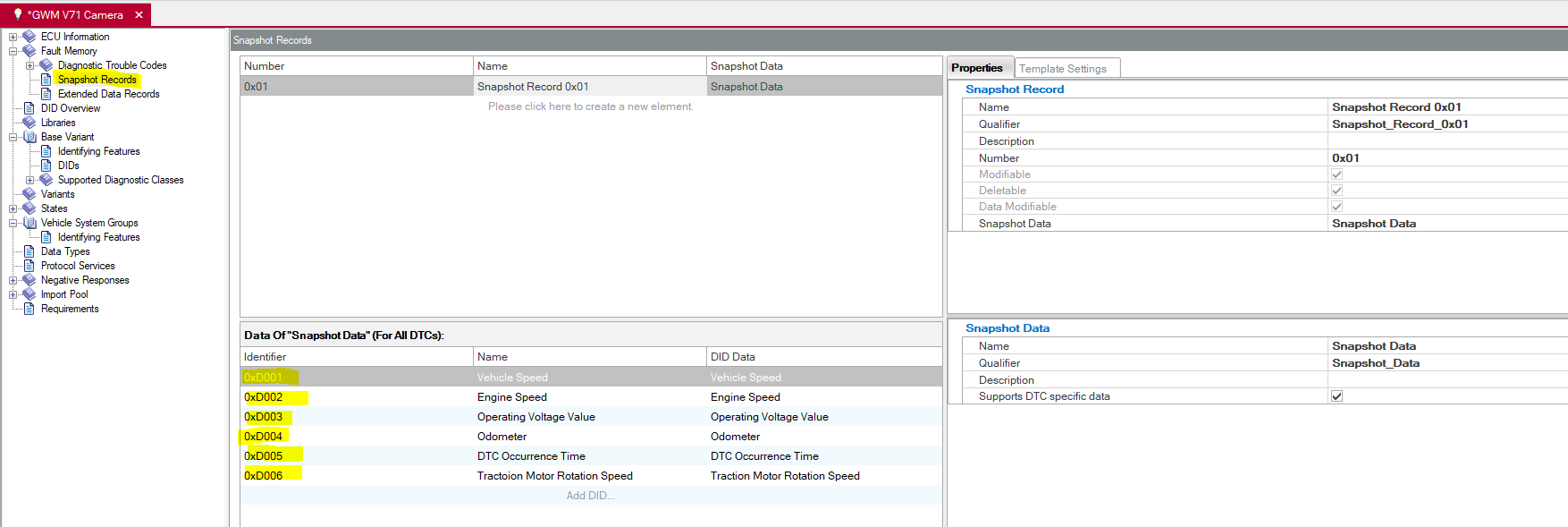
This ExtendedDataRecord Class nothing but DemExtendedDataRecord which comes from Cdd file



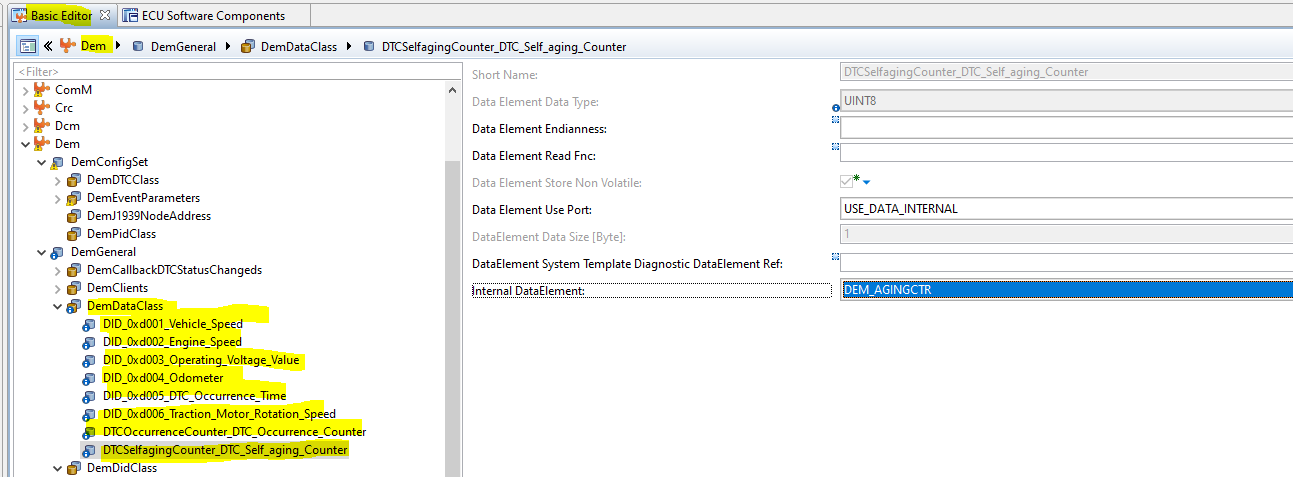
Below Snapshot and extended data we configured in CDD



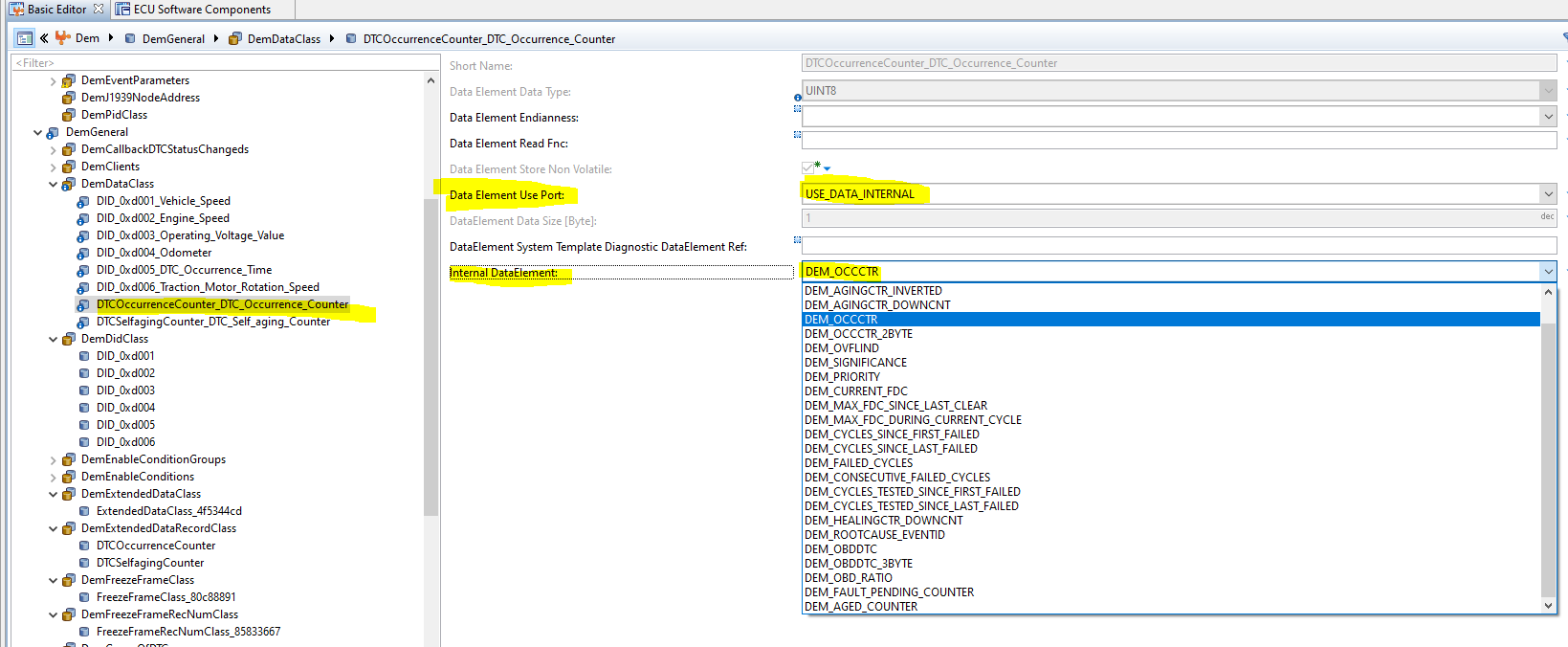


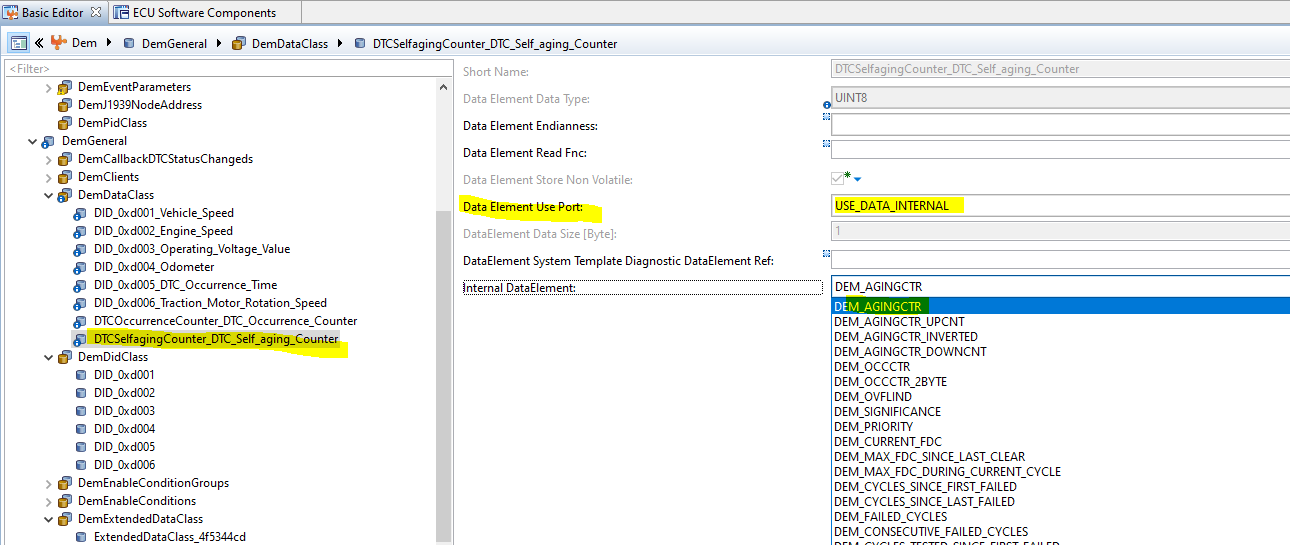


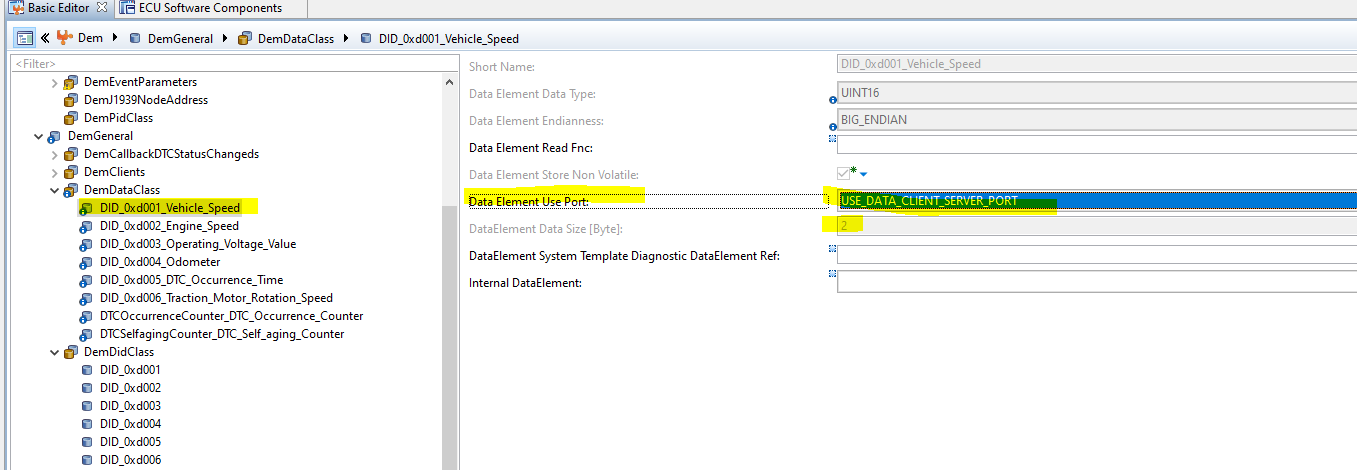
Once we import in Configurator below things configure automatically



In this freeze frame related DID we implement ,But extended data we configure ,check below

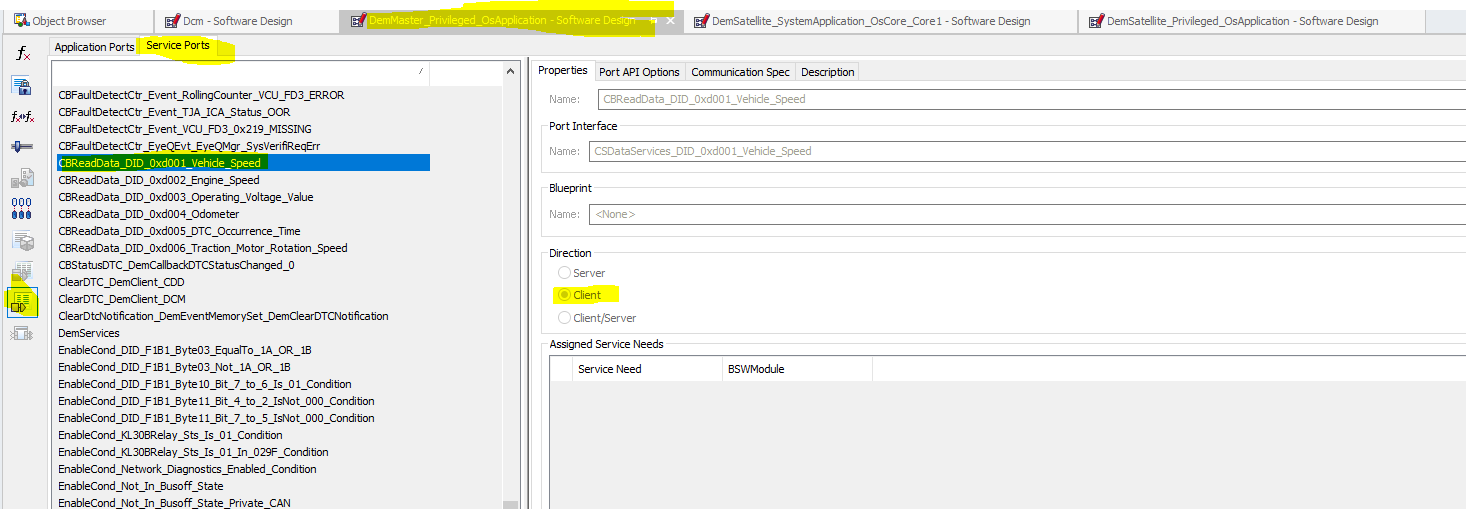


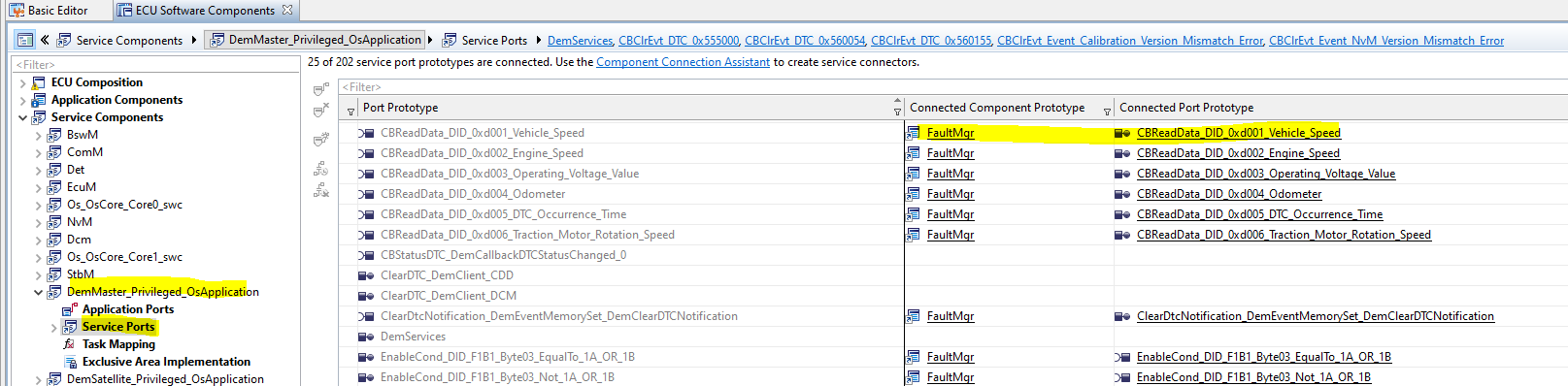






In DemMaster Freeze frame DID configured as service port(Client port)





Master port

