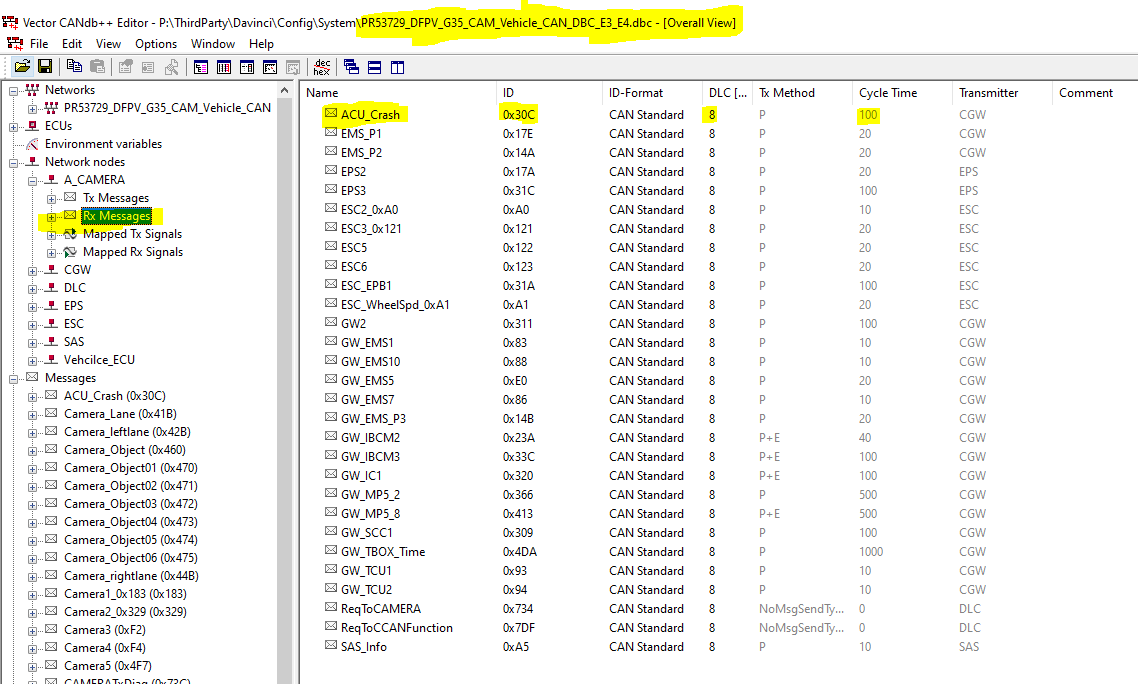
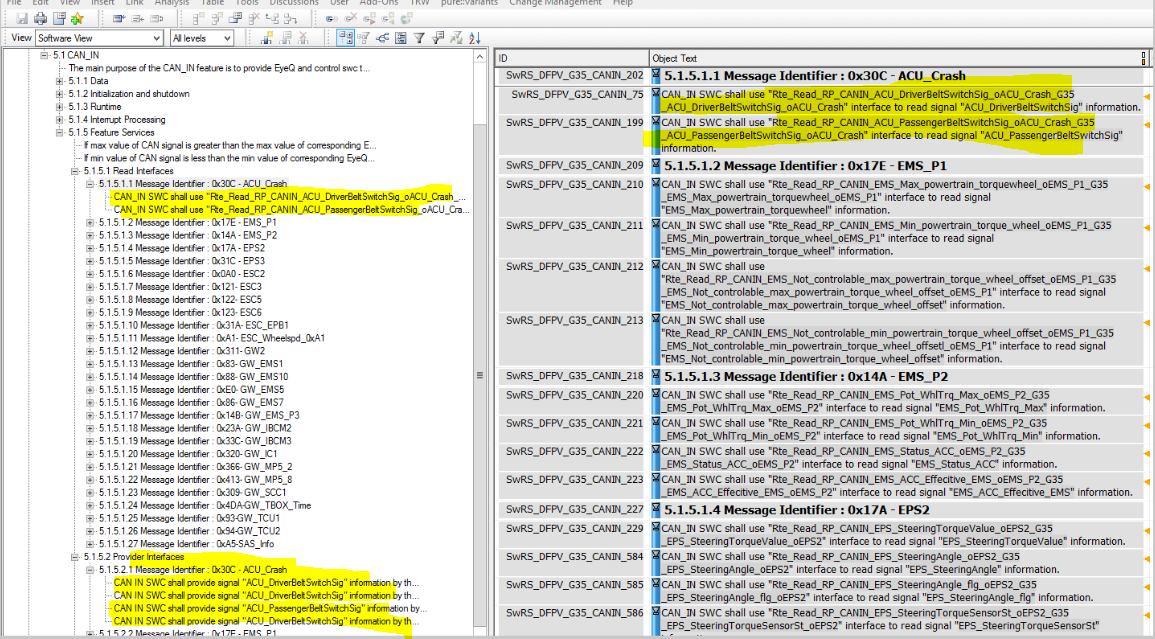
CHAP : 1

As per shown in DBC below are the Vehicle CAN Messages input to Camera ECU.



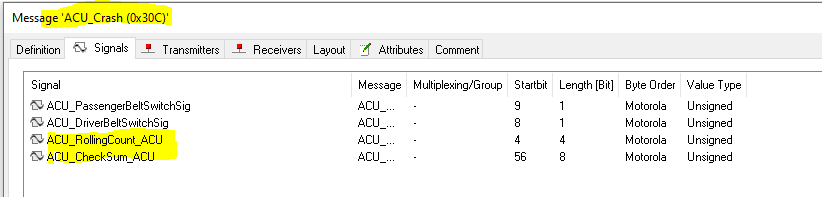
CHAP : 2

Consider message ACU\_Crash (0x30C) and check requirement for the same

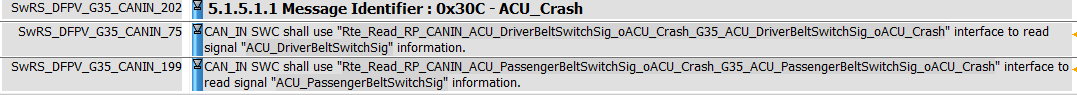


Message ACU\_Crash contains two signal and Rolling counter and checksum applicable for this Message .

Rolling counter and checksum information not given in CAN\_IN requirement but we need to take care of that in Callout function



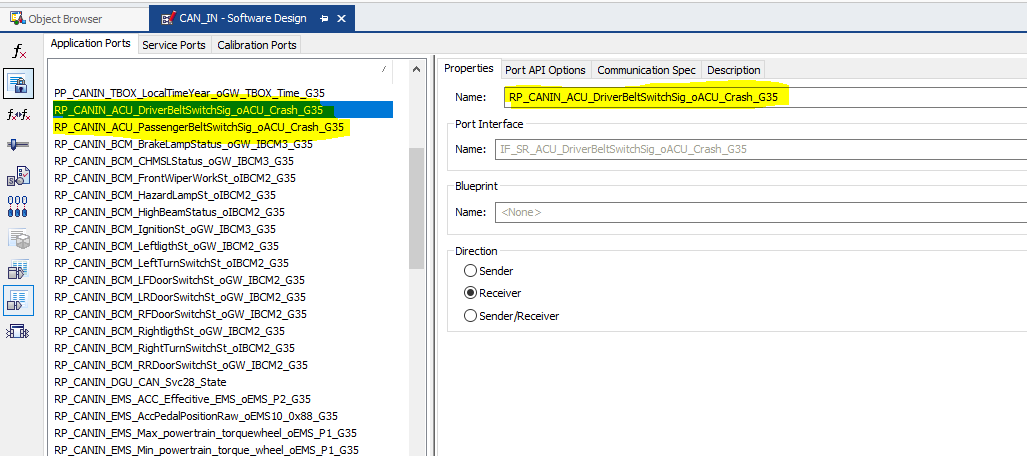
In DOORS which RTE variable signal information will read it mentioned. check below snap for the same.



So it’s CAN\_IN module responsibility to create Port as mentioned in requirement in SWC of CAN\_IN , For this port , Port Interface information provided by Arxml developer.

"Rte\_Read\_RP\_CANIN\_ACU\_DriverBeltSwitchSig\_oACU\_Crash\_G35\_ACU\_DriverBeltSwitchSig\_oACU\_Crash

Rte\_Read\_RP\_CANIN\_ACU\_PassengerBeltSwitchSig\_oACU\_Crash\_G35\_ACU\_PassengerBeltSwitchSig\_oACU\_Crash"

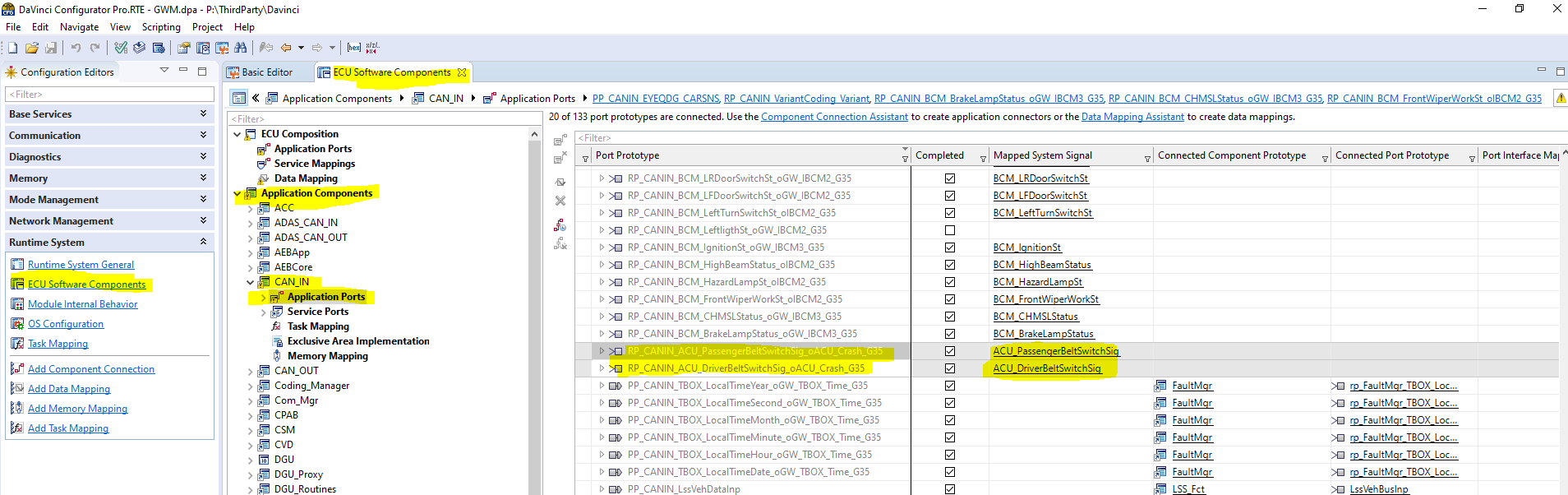


Right now port it is created in SWC ( CAN\_IN) which is above RTE layer so it’s Developer responsibility to provide this signal information ( COM layer) to SWC (Above RTE) by connecting it into the Configurator

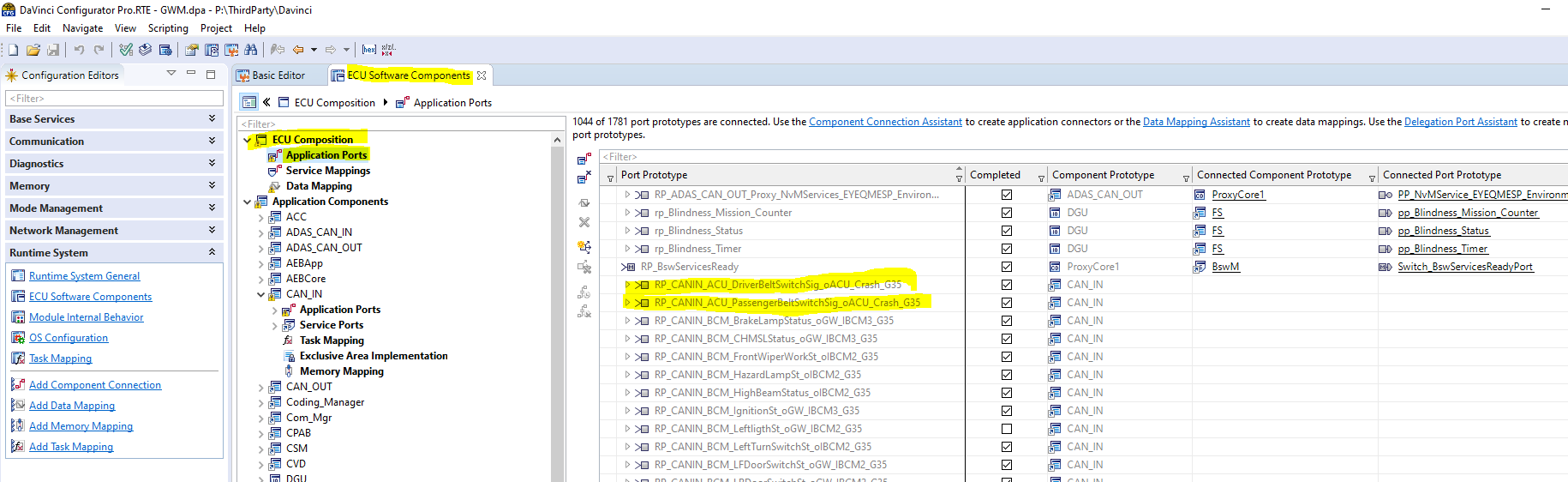
Check Below images for the same by connecting com signal to receive port in CAN\_IN

Check this port is connected in both the level 1) Application Components 2) ECU Composition

Application Component



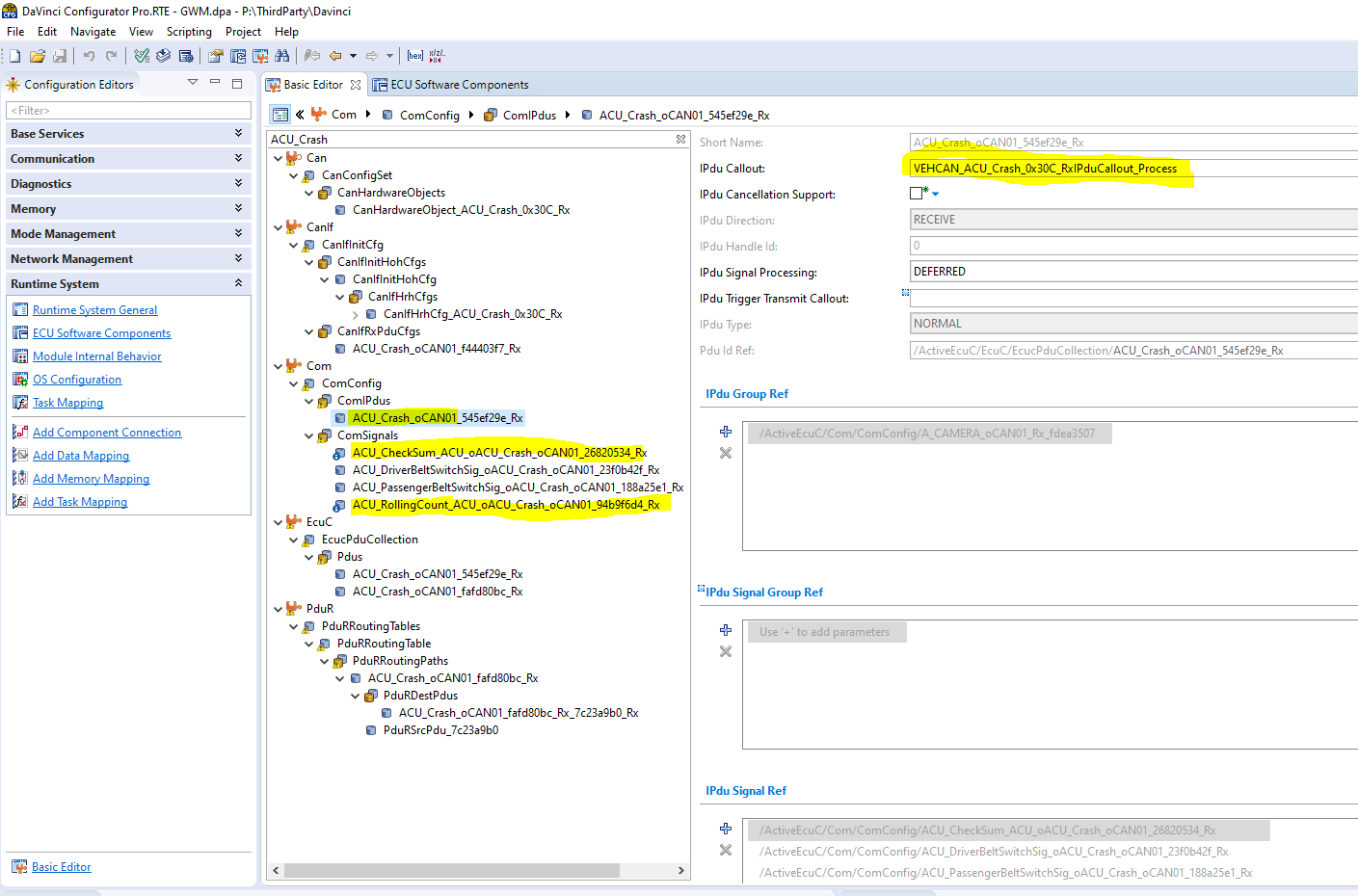
ECU Compostion



Now signal information available at port ,but how to read this and how to handle Rolling count and Checksum we will see in next chapter

CHAP : 3

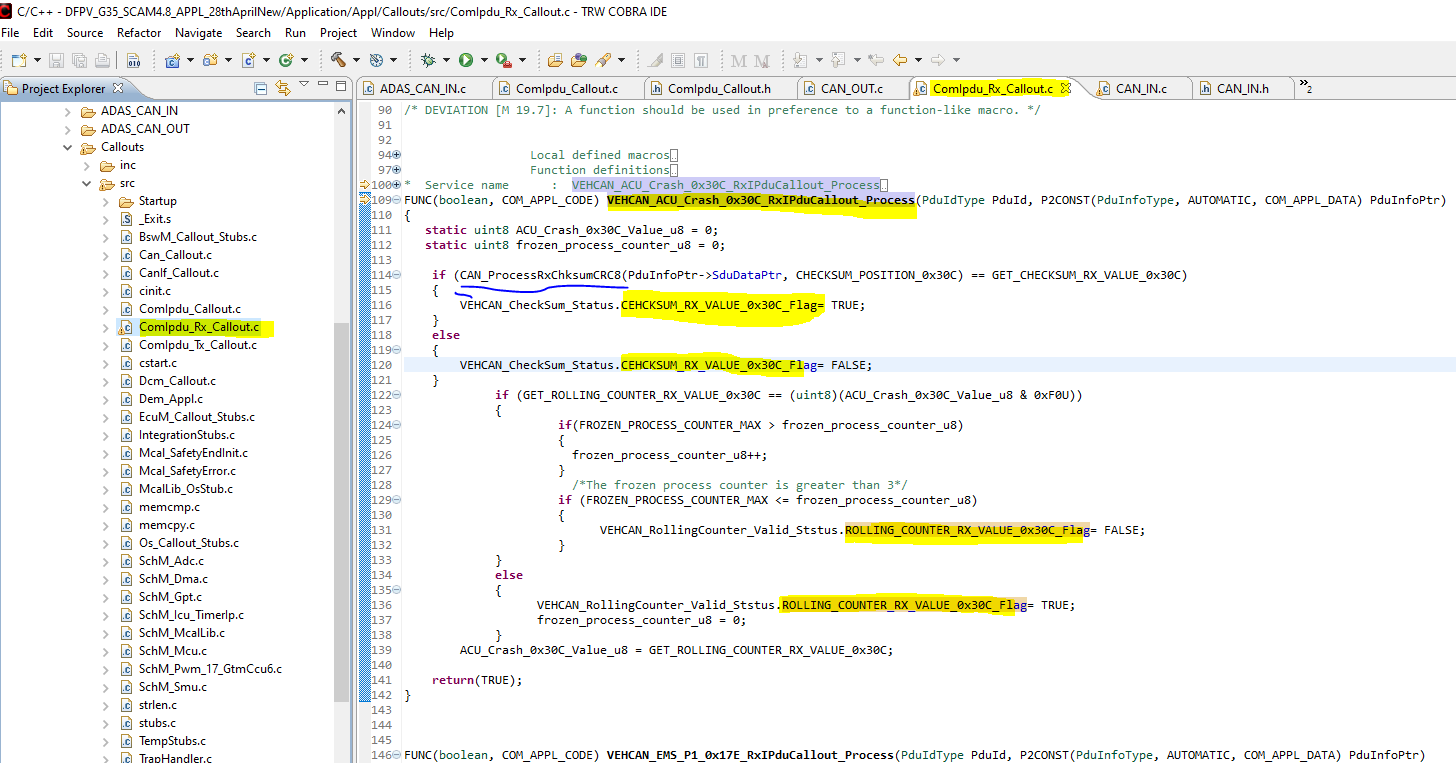
As seen in below image for message ACU\_Crash one IPdu Callout is mentioned : **VEHCAN\_ACU\_Crash\_0x30C\_RxIPduCallout\_Process ,** so this callout will implement as per developer needs



So in DF-G35 project this callout function is using to check CRC and Rolling count of that message(ACC\_Crash).

For all Vehicle CAN Rx message those having Checksum and Rolling counter has taken care inside that mentioned callout function in file ComIpdu\_Rx\_Callout.C

So check in below image Rolling counter(ROLLING\_COUNTER\_RX\_VALUE\_0x30C\_Flag) flag and checksum(CEHCKSUM\_RX\_VALUE\_0x30C\_Flag) flag, false or true set inside this callout function.



CHAP : 4

Reading of information and sending to interested SWC

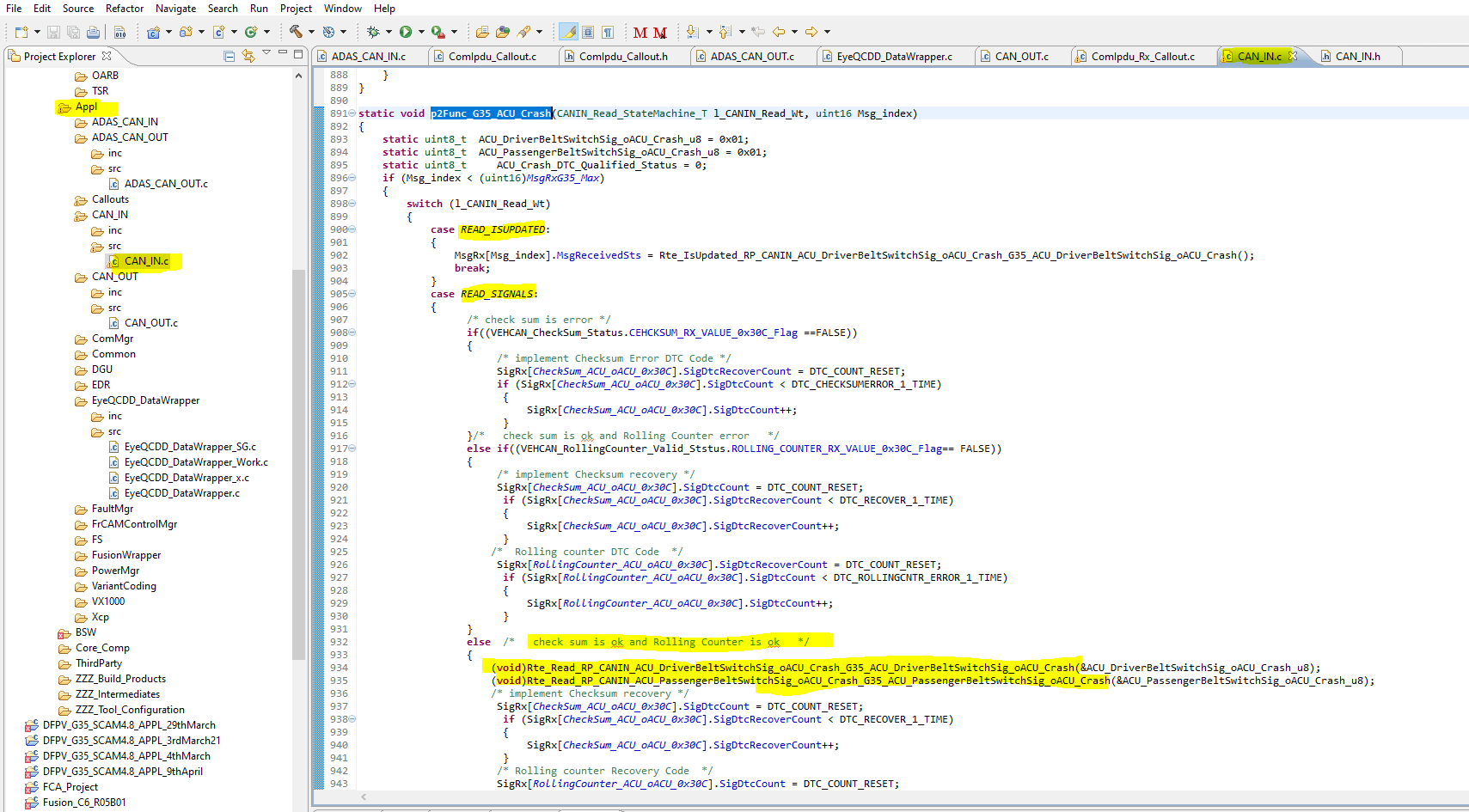
It’s developer responsibility to provide Vehicle CAN\_IN signal information to interested SWC if there is no error for that signal.

Before to Write/Read information, Checksum and Rolling need to check, which is set TRUE or FALSE in Callout function

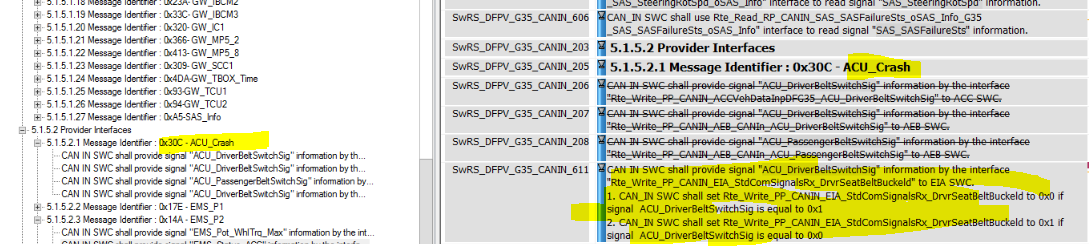
Check below implementation in CAN\_IN file

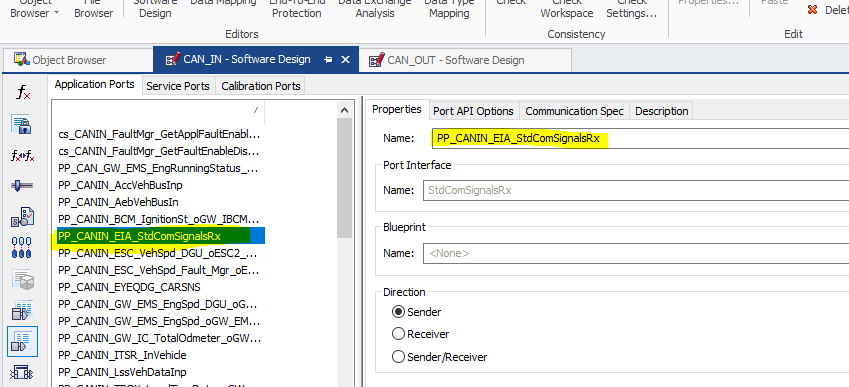
1. *READ\_ISUPDATED 2) READ\_SIGNALS* 3) *WRITE\_ACT\_DATA*

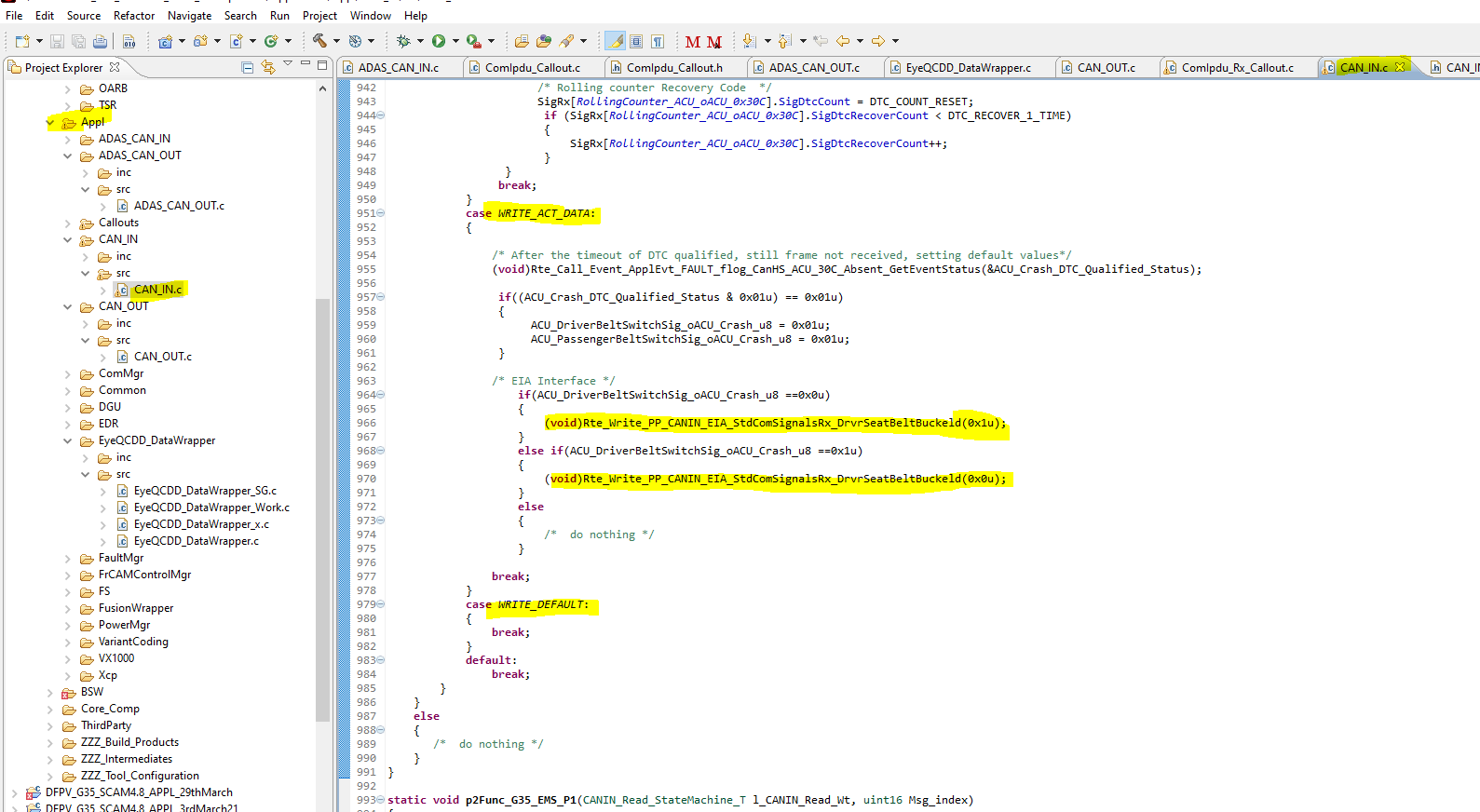
Check in below image line number 933 and 934 read will happen from COM layer



Check in below image line number 966 and 970 write will happen to which module interested in this. Here EIA interested





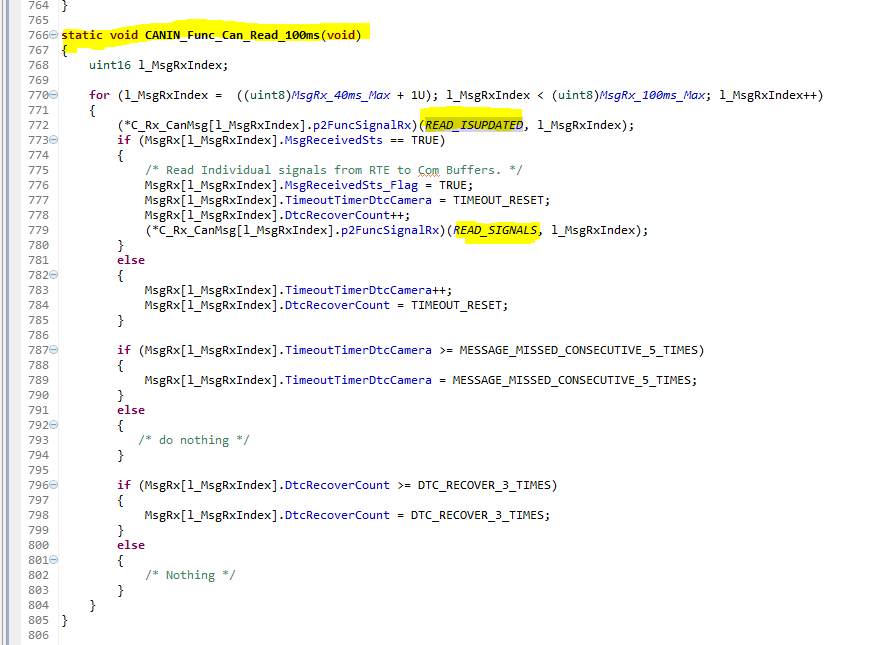


# READ\_ISUPDATED and READ\_SIGNALS

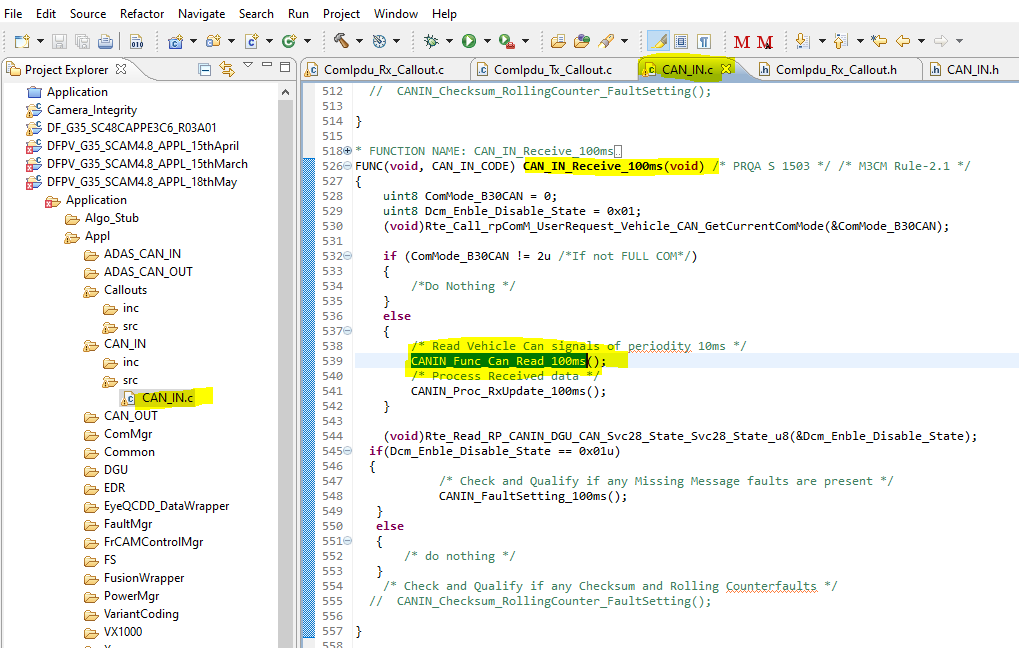
As shown below image signal read flag update and read of signal happening using this function.

[ Using this function “**CAN\_IN\_Receive\_100ms**(**void**)” we are calling to function **p2Func\_G35\_ACU\_Crash**(CANIN\_Read\_StateMachine\_T l\_CANIN\_Read\_Wt, uint16 Msg\_index)

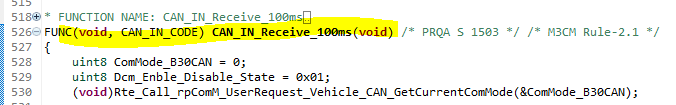
Line number 772 and line number 779 ]

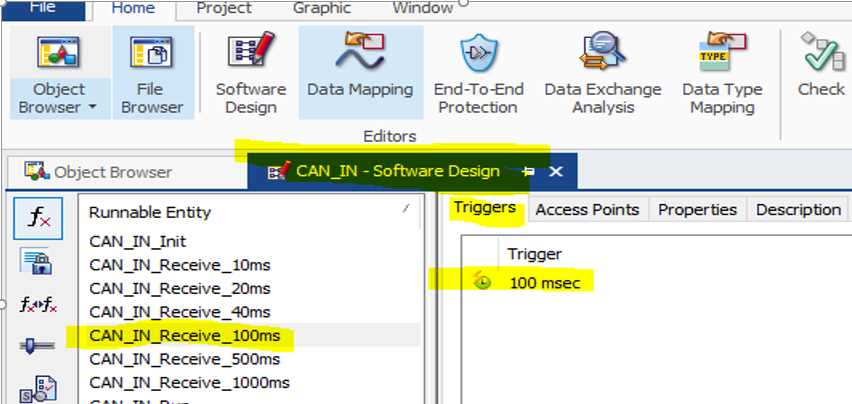


Above function is executing every 100ms . Check below pic for the same



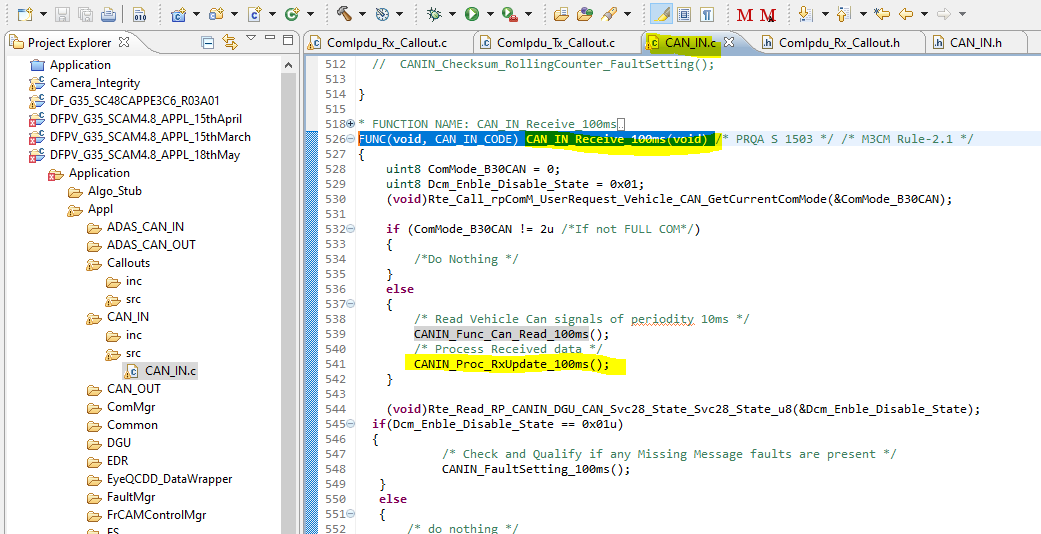
And parent of this function ‘FUNC(**void**, CAN\_IN\_CODE) **CAN\_IN\_Receive\_100ms**(**void**)’ mapped to developer

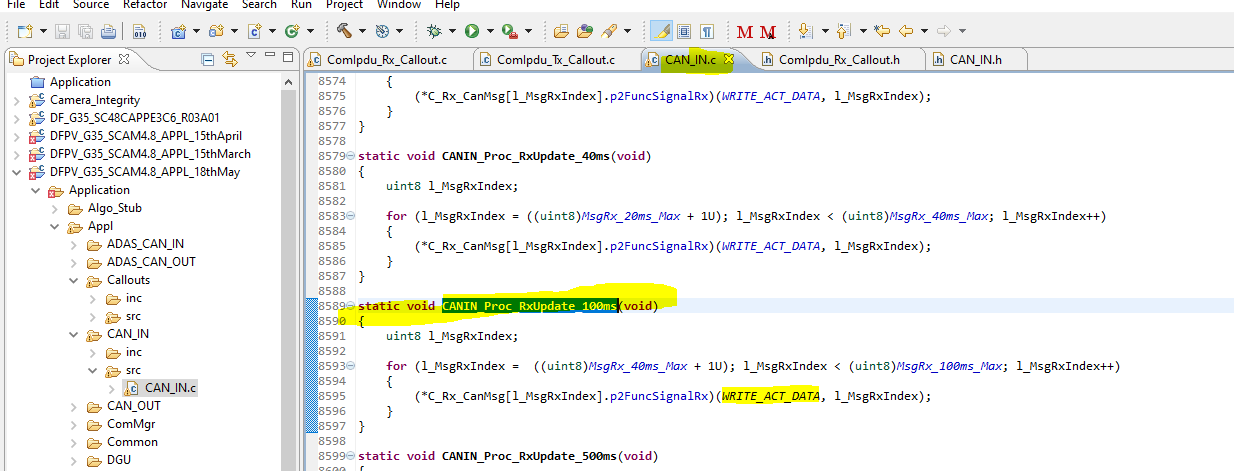




# WRITE\_ACT\_DATA

As shown below image write of signal happening using this function.





And parent of this function ‘FUNC(**void**, CAN\_IN\_CODE) **CAN\_IN\_Receive\_100ms**(**void**)’ mapped to developer which is shown previous page

