

Add-on II

Currently, RearECU uses CAPL code to implement a conversion of a light mode state to its physical value. The implementation is provided in the following file:

.\<ProjectName>\CANoe\RearECU.can

Move the light mode state conversion implementation from RearECU to MyECU.

The function 'on message msg_MyECU_2_Rear' shall be modified (see the picture below).

```
65 on message msg_MyECU_2_Rear
66 {
67     PutValue(env_LeftHeadlightSteering, this.sig_LeftHAngle);
68     PutValue(env_RightHeadlightSteering, this.sig_RightHAngle);
69
70     if (this.sig_LeftMode == 0)
71     {
72         PutValue(env_LeftHeadlightMode, 50);
73         PutValue(env_RightHeadlightMode, 50);
74     }
75     else if (this.sig_LeftMode == 1)
76     {
77         PutValue(env_LeftHeadlightMode, 70);
78         PutValue(env_RightHeadlightMode, 120);
79     }
80     else if (this.sig_LeftMode == 2)
81     {
82         PutValue(env_LeftHeadlightMode, 140);
83         PutValue(env_RightHeadlightMode, 200);
84     }
85     else
86     {
87         PutValue(env_LeftHeadlightMode, 255);
88         PutValue(env_RightHeadlightMode, 255);
89     }
```

Figure 1- The content of 'on message msg_MyECU_2_Rear' function (CAPL code)

1) The signals sig_LeftMode and sig_RightMode are 2b length. In order to send physical values (range from 0 to 255) from MyECU to RearECU the length of the signals shall be extended. To modify it use the following dbc file:

\\<ProjectName>\Config__InputDataBases\CAN.dbc

Is the signal naming still valid?

2) Update BSW and SWCs

2.1) Import the new dbc file (i.e. configure BSW)

2.2) Modify SWC design according to the new requirement

2.3) Implement the conversion logic and test ECU functionality

Hints:

- Modify algorithm of CtApAFSController SWC
- ECU functionality shall remain the same