



# Software-in-the-Loop using virtual CAN buses Virtualization

## Motivation

- 5 Series Transmission Software development
  - 7 Simulation software development
  - 7 In-vehicle software from supplier
  - 7 Different software variants
  - 7 :3:ode ', and .oded,  
auto .ode from TargetLink()
- 5 Frontload development tasks
  - 7 Debugging of series transmission



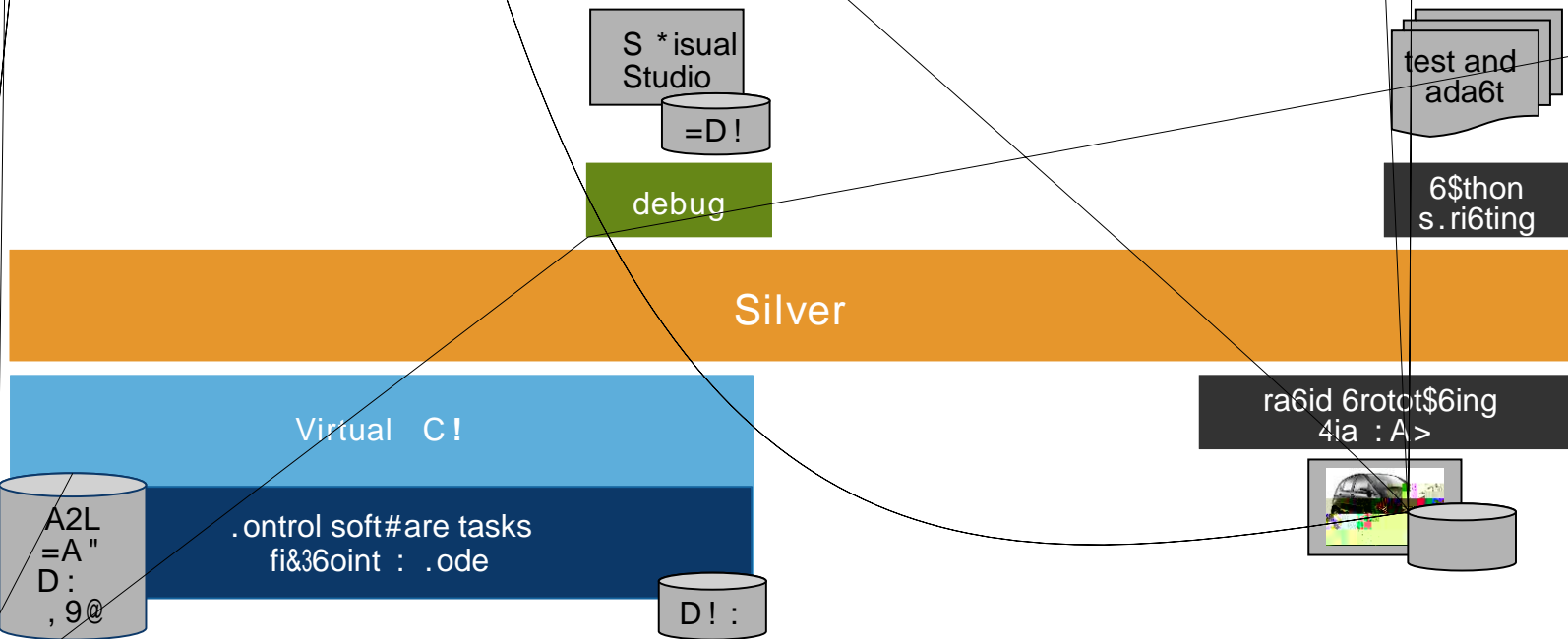
# Software-in-the-Loop using virtual CAN buses Virtualization

5 Software-in-the-Loop (SiL)  
7 Integration of the Virtual ECU



# Software-in-the-Loop using virtual CAN buses Sil4er

- 5 Sil4er from - Troni. +mb,
- 7 Soft#are3in3the3Loo6 'SiL( simulation en4ironment
- 7 All rele4ant automoti4e standard formats su66orted 'A2L, =A ", D! : (
- 5 )A\* has alread\$ e&6erien.e #ith Sil4er for 2 \$ears

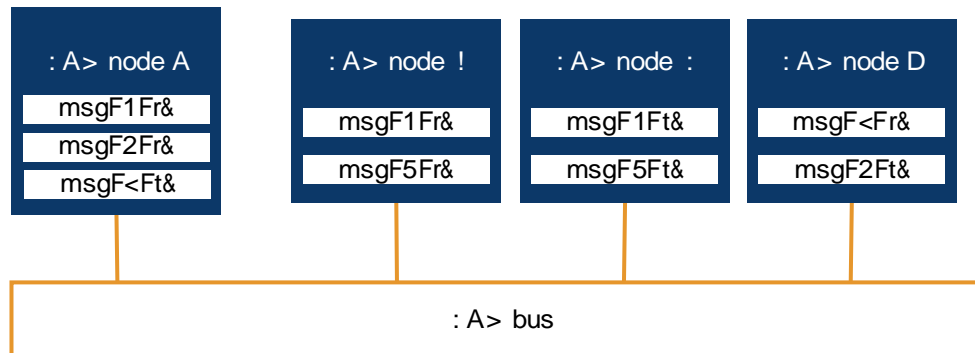




# Software-in-the-Loop using virtual CAN buses

## : A> !asi.s

- 5 ! us) Bne transmitter 3 multi6le re .ei4ers
- 5 >ode) t\$6i .all\$ one 9 : ; , transmits or re .ei4es messages
- 5 message) u63to E !\$te data, .\$.li. or e4ent3based, 6riorit\$
- 5 Signal) 6a .ked into a message, s .aled b\$ gain1offset '13C? bit(
- 5 D! : 8ile) s6e .ifies : A> bus, es6e .iall\$ message stru .ture and timing

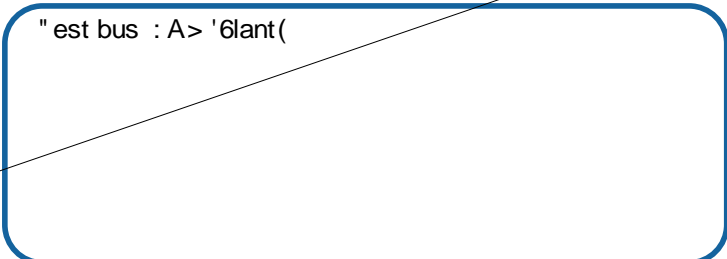


# Software-in-the-Loop using virtual CAN buses

## : A> !asi.s) rest bus

---

- 5 SiL simulation fo. used on one 9 : ;
  - 7 D! : defines #hole bus
  - 7 >ot all nodes! messages are needed in simulation
  - 7 9mulation filters b\$
    - 5 node names
    - 5 bla.k listing or #hite listing of messages



> rest bus : A> '6lant(

# Software-in-the-Loop using virtual CAN buses

## Signals in .ommon SiL

```

5 :ommon SiL signal rooting '#ithout :A>(
7 *irtual 9 : ;1 8un.tion soft#are 'Sil4er : A=)(
5 "emo4e :A> .ode
5 define Silver &'O for .ode 4variable
'gain and offset manual$(
7 odel 'Sil4er Simulink blo.k set(
5 define Silver &'O for Simulink signal

```

```

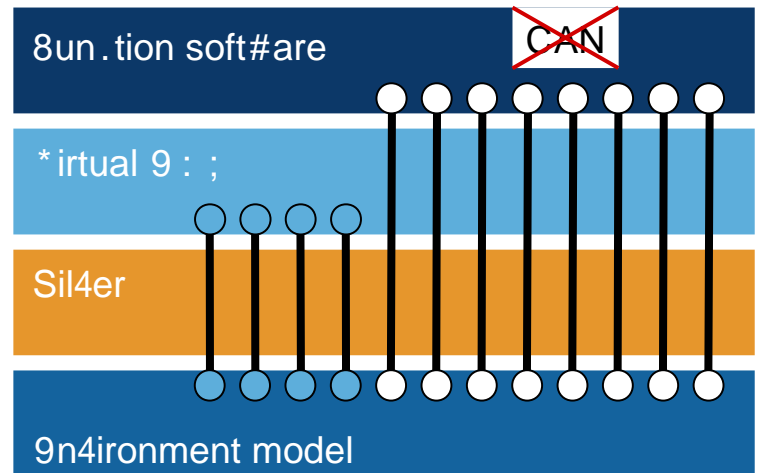
5 Sil4er
7 dete.ts Sil4er )1B signals b$name
7 .o6ies information automati.all$ at
begin1 end of simulation ste6
7 .onne.tion data t$6e) double

```

```

5 Summar$
7 8un.tion soft#are :A> .ode is b$6assed
7 T$6i.all$ man$ signals to be set u6

```



Legend  
 ● =in Sil4er )1B  
 ○ :A> Sil4er )1B



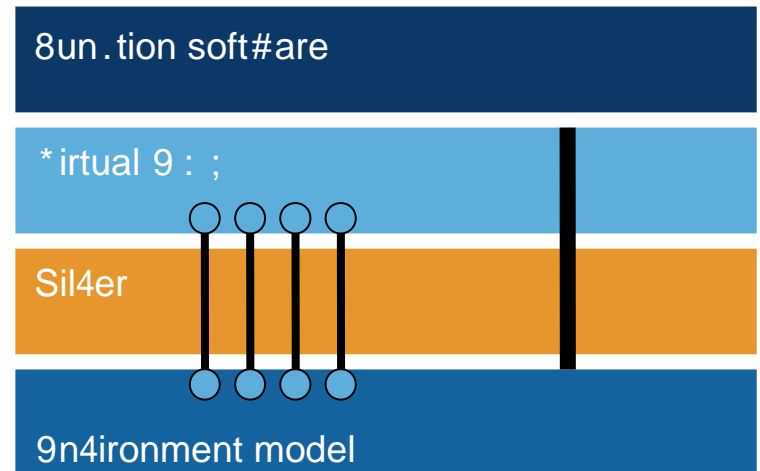


# Software-in-the-Loop using virtual CAN buses

## Signals in SiL #ith : A> '2(

### 5 Summar\$

- 7 8un .tion soft#are : A> .ode in simulation,  
.an be debugged and tested
- 7 "edu .ed .ommuni .ation setu6 effort
  - 5 Less manual definitions
  - 5 Automati . s .aling from1 to : A>  
message
- 7 ;se of D! : #ork 6rodu .t
- 7 :ar .om6arable .ommuni .ation beha4iour



# Software-in-the-Loop using virtual CAN buses ;sing Sil4er : A> emulationD : ode

5 Sil4er A=) 2.% defines : fun.tions forD

7 : onfigure : A> bus or busses b\$ D! : file or b\$ single messages

7 Start1 sto6 : A> emulation

7 : he.k ne# message re.ei4ed

7 Transmit1 re.ei4e message 'E b\$te data(

7 Transmit1 re.ei4e signal1 4variable 'from a message, ra# or s.aled(

7 ani6ulate messages for .ounter1 : " : '.all3ba.k dll(

\*irtual 9 : ;

```
...
// can bus ID
1, // can bus ID
"trans_dhc" // file name
...
"ignore.txt", // ignore file name
...
0x0 // CAN message
...
return 1;
}
```

# Software-in-the-Loop using virtual CAN buses ; using Sil4er : A> emulation) Simulink

5 Sil4er : A> block set

7 ! us message setu6 'Sil4er : A>msg ! us(

: onfigure one : A> bus b\$ D! : and node names

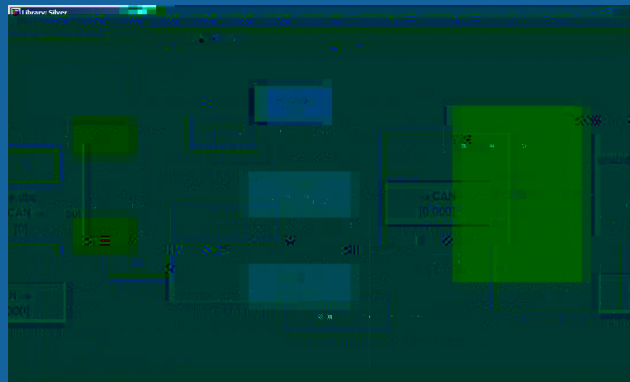
9nable1 disable node b\$ Simulink in6ut bus

Transmit messages from Simulink in6ut bus

But6ut re.ei4ed messages to Simulink out6ut bus

7 Single message setu6 'Sil4er : A>bus : onfig, Sil4er : A>bus " ead,  
Sil4er : A>bus I rite(

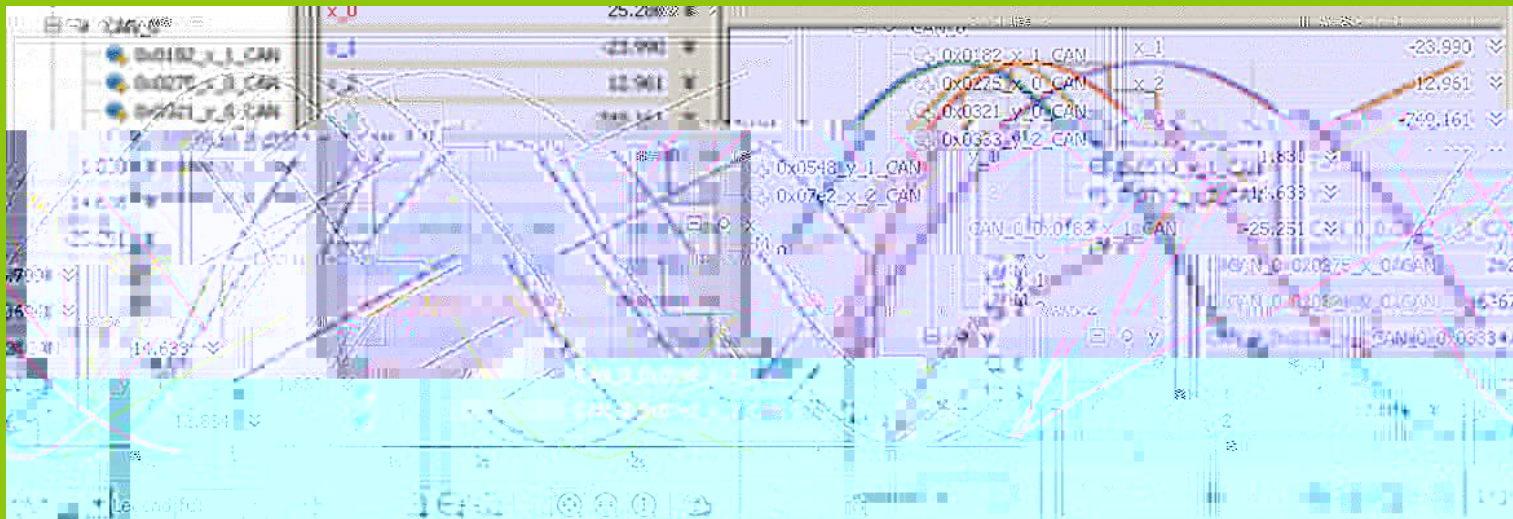
9n4ironment model



# Software-in-the-Loop using virtual CAN buses ( ; sing Sil4er : A> emulationD ; ser + ; )

- 5 See .onne .tion state of : A> messages
- 5 9as\$ a . .ess to : A> signals for 6lotting1 debugging

Sil4er + ; )

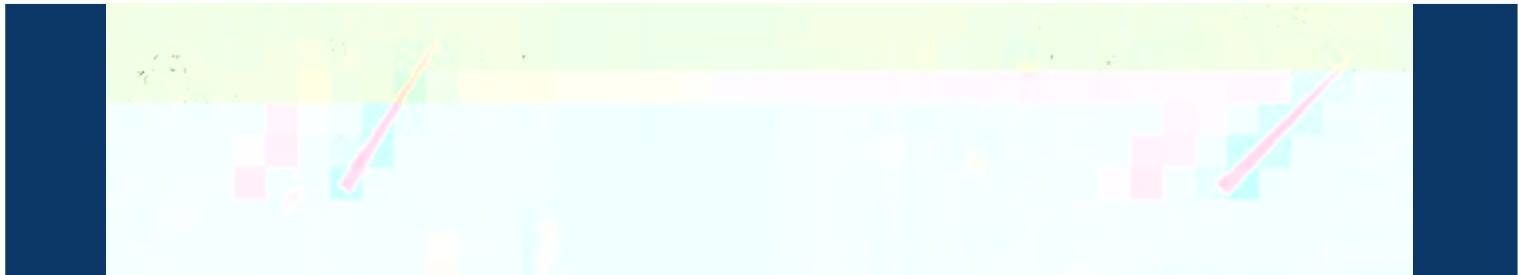


# Software-in-the-Loop using virtual CAN buses

## Summary

---

- 5 Network communication is an important part of system design
- 5 Using CAN emulation in SiL simulations
  - 7 drastically reduces the effort for defining SiL communication setups
  - 7 improving the consistency of definitions
  - 7 adds additional simulation aspects: signal scale, communication timing
  - 7 enables tests of the CAN-related software parts which are up to now bypassed



# Thank You

---

Dr. Thomas Liebezeit  
)A\* +mb ,

: arnotstraKe 1, 1%5EG ! erlin  
Telefon L?H <% <HHGE3H%21

thomas.liebezeitMia4.de

###ia4..om