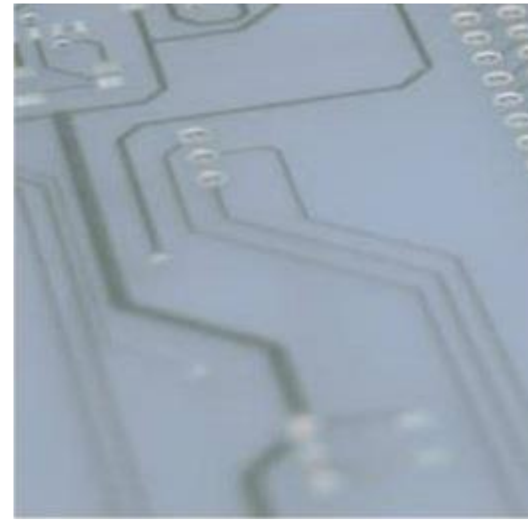
J1939

H.-J. Schleißheimer
Soft- und Hardware
Entwicklung GmbH



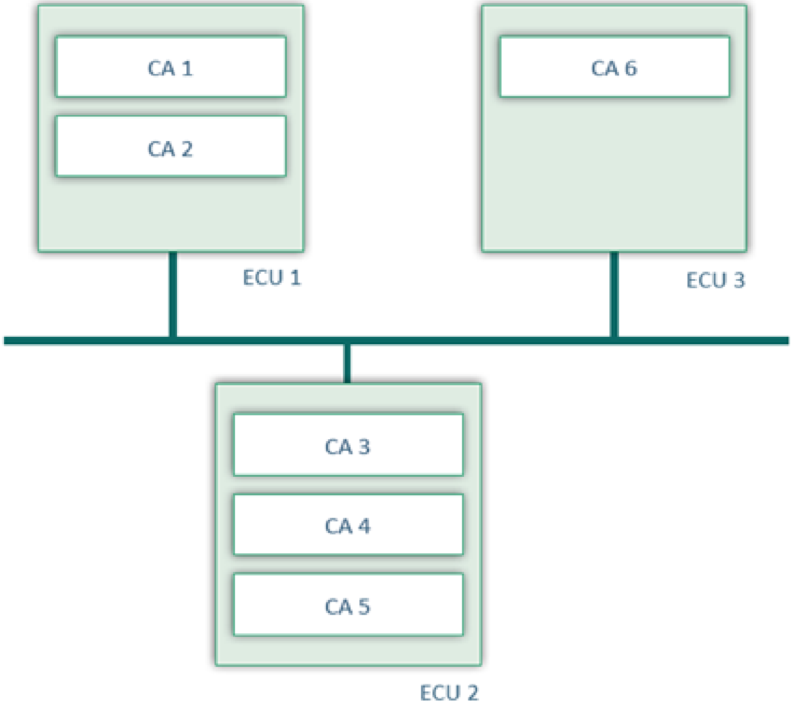
Holger Dahinten

J1939 – Introduction

- J1939 is a set of standards defined by Society of Automotive Engineers (SAE)
- It is mainly used in heavy-duty vehicles such as trucks and buses, mobile hydraulics
- Higher-layer protocol built on CAN
- The speed is nearly always 250 kbit/s or 500 kbit/s

J1939 – CA (Controller Application)

- The Controller Application is Software part of an ECU. An ECU can contain one or more CAs. Each CA has a unique address and an associated device name. Each message sent by a CA contains this source address.
- A parameter group is like a CAN message.
- A CA can have multiple PGs
- A PG consist of signals so called parameters.

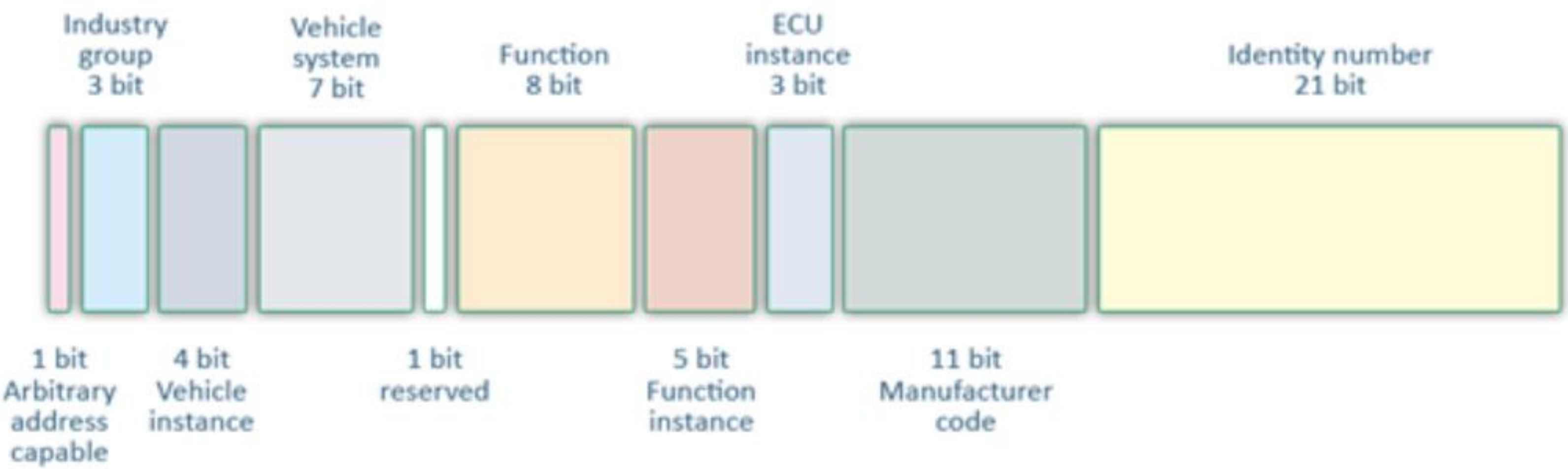


J1939 – Addressing

- There are 256 possible addresses
- 0..127 Used for CA's with preferred addresses and defined functions.
- 128..247 Available for all CA's
- 248..253 Used for CA's with preferred addresses and defined functions
- 254 Null
- 255 Global

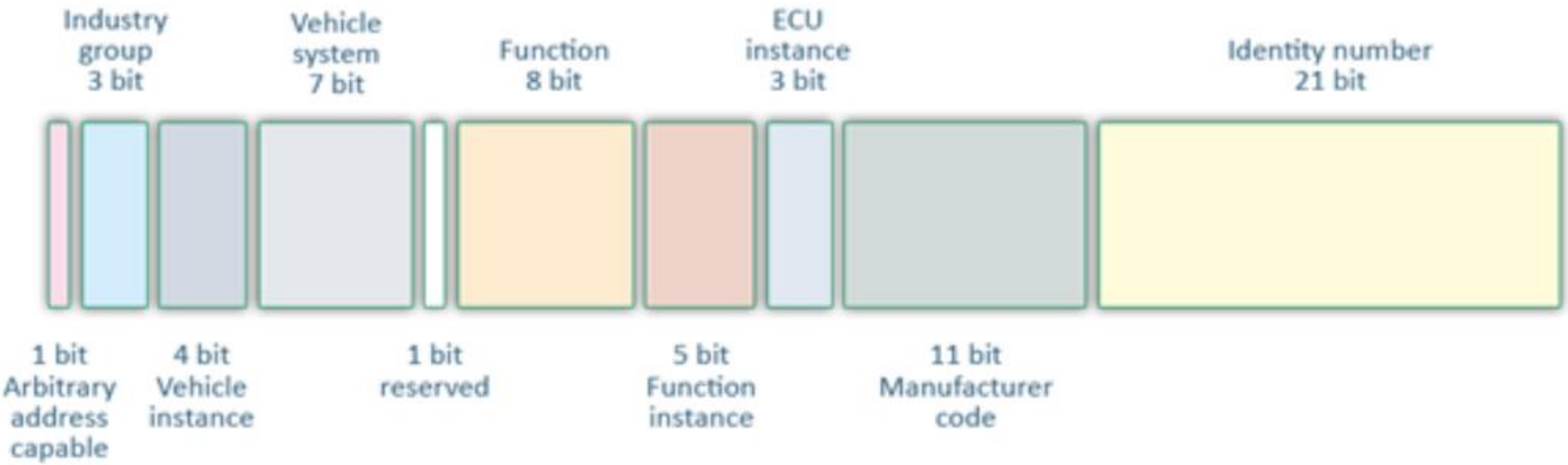
J1939 – Device name (1/2)

- J1939 defines device names, which are each represented by a 64-bit (8-byte) label and are used to identify the device and its function.
- The device name is divided into various elements, some of which have dependencies.
- The independent fields include the "Industry Group" and the "Manufacturer Code".



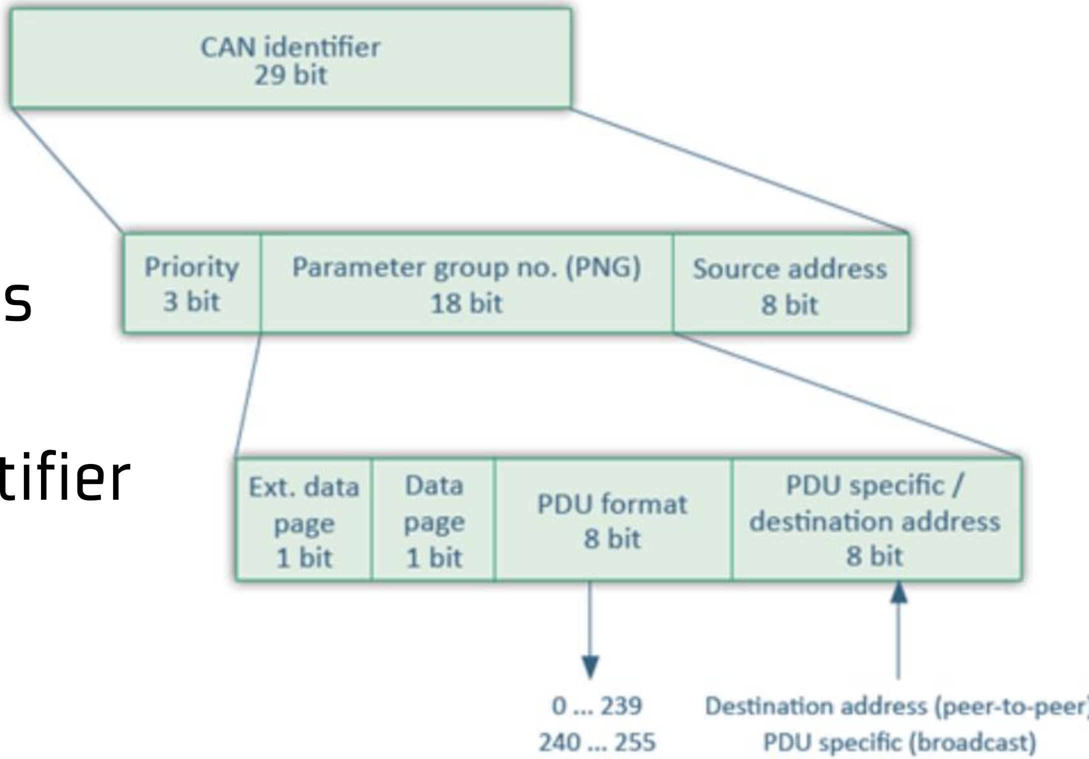
J1939 – Device name (2/2)

- The functions required in the network are defined via the industry group
- The manufacturer code must be applied for and assigned by SAE. With this manufacturer code and the additional identity number, the entire name of a device is unique worldwide.
- The function instance is required if several CAs have the same function.



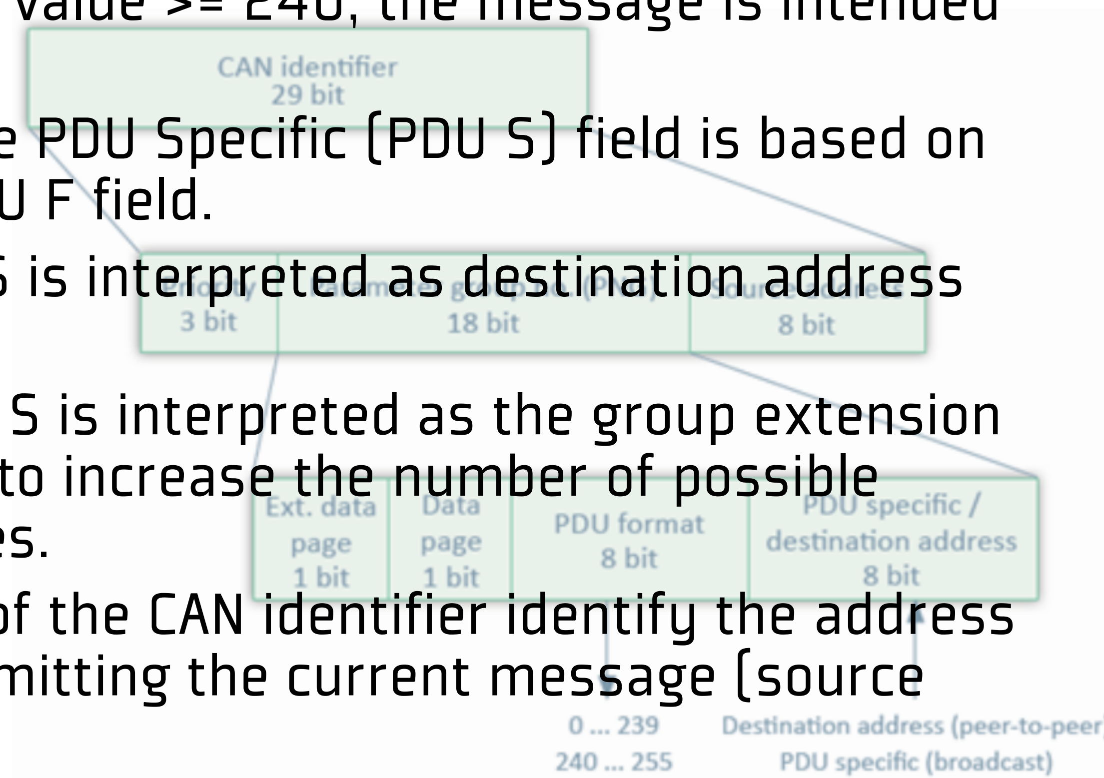
J1939 – CAN ID (1/3)

- J1939 messages are built on the CAN 2.0B specification and make specific use of "extended frames".
- These use a 29-bit identifier
- J1939-21 defines the fields within this 29-bit identifier



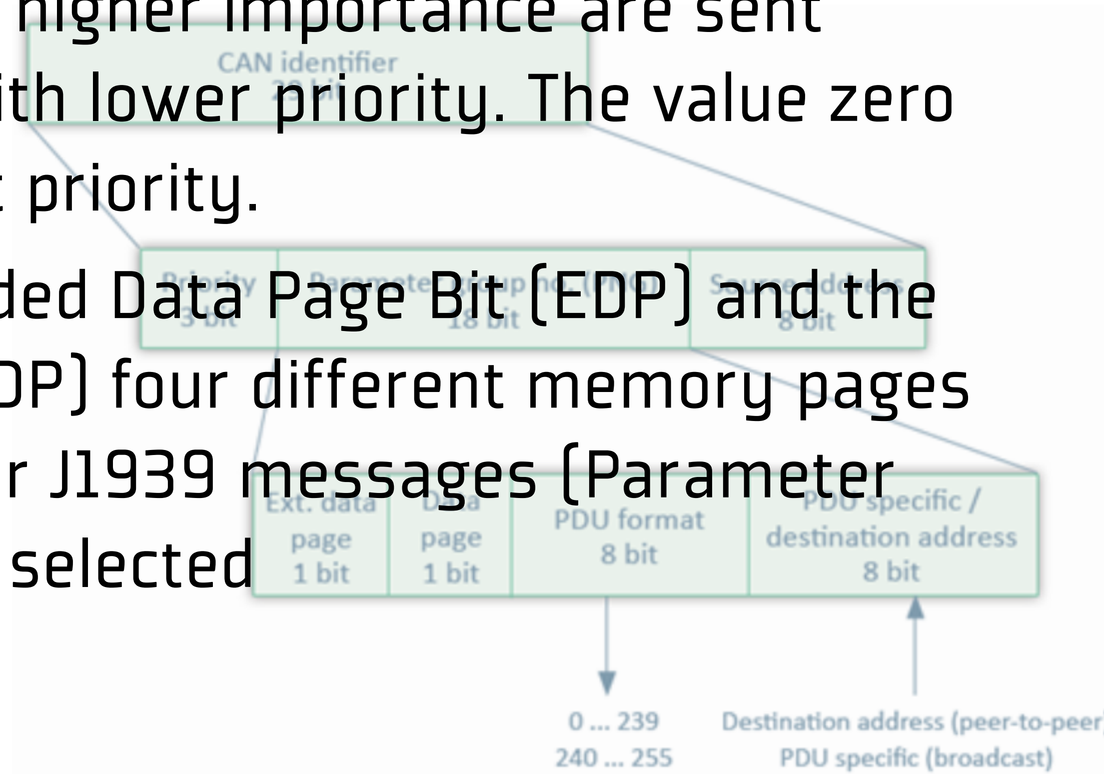
J1939 – CAN ID (2/3)

- The PDU Format field (PDU F) defines whether the message is intended for a specific device in the network or for the entire network. If the value of PDU F < 240, a specific device is addressed, if the value >= 240, the message is intended for all devices.
- The definition of the PDU Specific (PDU S) field is based on the value of the PDU F field.
- PDU F < 240: PDU S is interpreted as destination address field.
- PDU F >= 240: PDU S is interpreted as the group extension field which is used to increase the number of possible broadcast messages.
- The last eight bits of the CAN identifier identify the address of the device transmitting the current message (source address field).



J1939 – CAN ID (3/3)

- The first three bits define the priority of the message on the network and ensure that messages with higher importance are sent before those with lower priority. The value zero has the highest priority.
- With the Extended Data Page Bit (EDP) and the Data Page Bit (DP) four different memory pages (Data pages) for J1939 messages (Parameter Groups) can be selected



J1939 – Address Claiming

- Before a CA uses an address, it must claim it in the network. This procedure is called "address claiming" (ACL).
- Here, the unique device name is used to resolve conflicts in address assignment. The smaller the numerical value, the higher the priority.
- The CA sends an "Address Claim PGN" (ACL, PGN 00EE00h) when it starts and waits a specified time for a response. If no other CA claims this address within this time, it can start with normal communication.

J1939 – Transport protocol

H.-J. Schleißheimer
Soft- und Hardware
Entwicklung GmbH

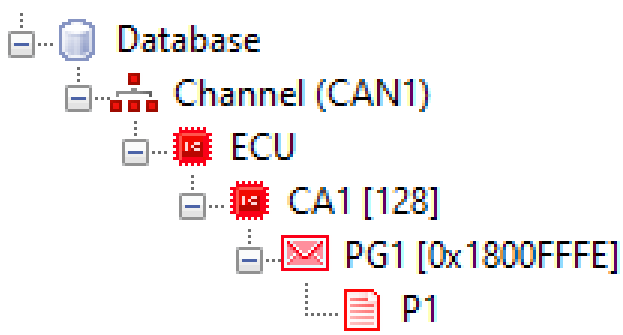
- **Peer to Peer**
  (CMDT, Connection Mode Data Transfer)
  In subscriber-oriented message transmission (peer-to-peer), the destination address is specified. The data is directed to a specific subscriber and is confirmed.
- **Broadcast**
  (BAM, Broadcast Announce Message)
  Global transmissions are communications without confirmation (no flow control). In this case, the recipients of the message are not known to the sender.

J1939 – Overview of CanEasy Demo

- Create simple database
- Address claim
- Deactivate Address claim
- Update of Source Address
- Update of Destination Address
- Send a PG
- Setup NAME of CA
- Setup PGN
- PG Request
- Change SourceAddress for PG Request
- Use Transport Protocol

J1939 – Create simple database

- Create CAN channel
- Create control unit
- Create CA (Controller Application) "CA1"
- Create parameter group (message) "PG1"
- Create parameter (signal) "P1"



J1939 – Address claim

- Open trace (default) and Tools->J1939 claimed addresses window
- Start simulation
  -> Address claim is shown in both windows

New trace 1

Timestamp	C	Identifier	Name	Dir	D	Data	Channel
00000.0780	CAN1	0x18eeff80	AC (Address Claimed)	Tx	8	00 00 e0 ff 00 ff fe 80	Channel

J1939 claimed addresses

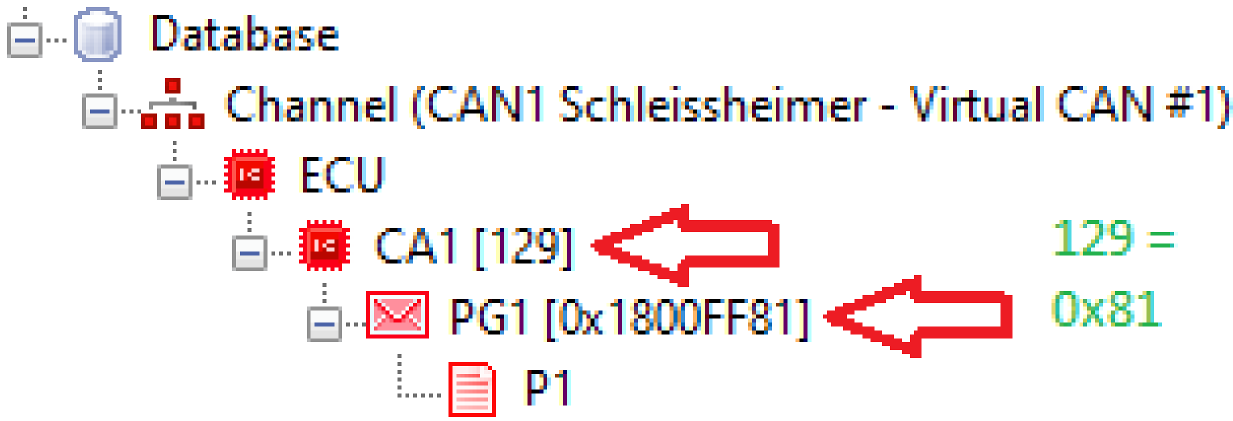
Cha...	Address	NAME	AAC	IndustryGroup	VehicleSystemInstance	VehicleSystem	Function	FunctionInstance	EcuInstance	ManufacturerCode	IdentityNumber
Channel	128	80feff00ffe00000	true	0	0	127	255	0	0	2047	0

J1939 – Deactivate Address claim

- Set Auto to False for CA1 and restart simulation
  -> No address claim is sent
- Set Auto to True and restart simulation
  -> Address claim is sent
- Set DefaultAddress to 255 and restart simulation
  -> No address claim is sent
- Set DefaultAddress back to 128
  -> Address claim is sent again

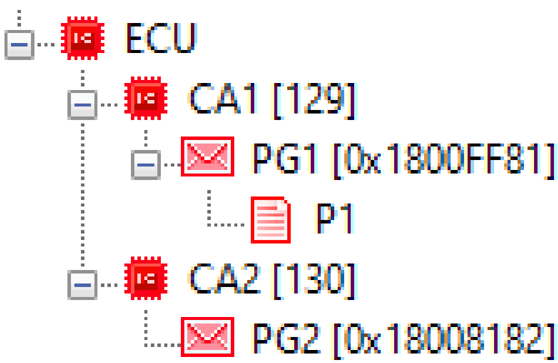
J1939 – Update of Source Address

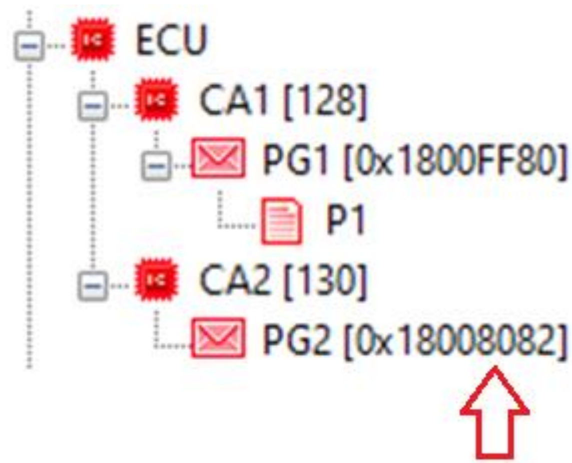
- Change DefaultAddress of CA1 to 129
  -> New address claim is sent
  -> Source address of PG(s) gets updated



J1939 – Update of Destination Address

- Create another CA and PG (CA2, PG2, DefaultAddress 130)
- Open "Parameter group destination editor" for new PG
- Choose CA1
  -> Destination address gets inserted into PG2
  -> CA1 property PGRefsUsingDA points to PG2
- Change DefaultAddress of CA1 to 128 (0x80)
  -> Destination address of PG2 gets updated





J1939 – Send a PG (message)

H.-J. Schleißheimer
Soft- und Hardware Entwicklung GmbH

- Double click on PG1
  -> auto generated panel opens
- Changing parameter P1
  -> PG1 gets transmitted
- Change TransmissionMode to CyclicAndSpontan
- Enter CycleTime 1000 for PG1
  -> PG gets transmitted each 1000 ms

New trace 1

Timestamp	C	Identifier	Name	Dir	D	Data	Channel
00010.9250	CAN1	0x1800ff82	PG1	Tx	8	5c 00 00 00 00 00 00 00	Channel
00011.2820	CAN1	0x1800ff82	PG1	Tx	8	5c 00 00 00 00 00 00 00	Channel
00011.3730	CAN1	0x1800ff82	PG1	Tx	8	8f 00 00 00 00 00 00 00	Channel
00012.0130	CAN1	0x1800ff82	PG1	Tx	8	5c 00 00 00 00 00 00 00	Channel
00012.2820	CAN1	0x1800ff82	PG1	Tx	8	5c 00 00 00 00 00 00 00	Channel
00012.5740	CAN1	0x1800ff82	PG1	Tx	8	8f 00 00 00 00 00 00 00	Channel
00012.5810	CAN1	0x1800ff82	PG1	Tx	8	c2 00 00 00 00 00 00 00	Channel
00013.2820	CAN1	0x1800ff82	PG1	Tx	8	c2 00 00 00 00 00 00 00	Channel

J1939 – Setup Name of CA

- Mapping of NAME to NAME_xxx parameters works in both directions
- Select CA1
  -> List view shows different NAME properties
- Change NAME_IndustryGroup to 1
  -> NAME property gets updated
  NAME = 0x80feff00ffe00000
  NAME = 0x90feff00ffe00000
- Change NAME property back to 0x80feff00ffe00000
  -> NAME_IndustryGroup returns to value 0

J1939 – Setup PGN

- Select PG1
  -> List view shows different MsgId properties
- Set MsgId_PGN_PDUFormat to 1
- Set MsgId_PGN_PDUSpecific to 2
  -> MsgId_PGN shows 0x100 (256) because "PDU specific" is destination address.
  (Hint: Use Value-Editor to change property display type to hex)
- Set MsgId_PGN_PDUFormat to 240 (0xf0)
  -> MsgId_PGN shows 0xf002 (61442) because "PDU specific" is part of the PGN now.
- Set MsgId from 0x18f00282 to 0x18f10382
  -> MsgId_PGN_PDUFormat gets updated to 241
  -> MsgId_PGN_PDUSpecific gets updated to 3
- Change MsgId_PGN_DataPage and MsgId_PGN_ExtendedDataPage
  -> MsgId_PGN and MsgId gets updated

J1939 – Send PGN Request

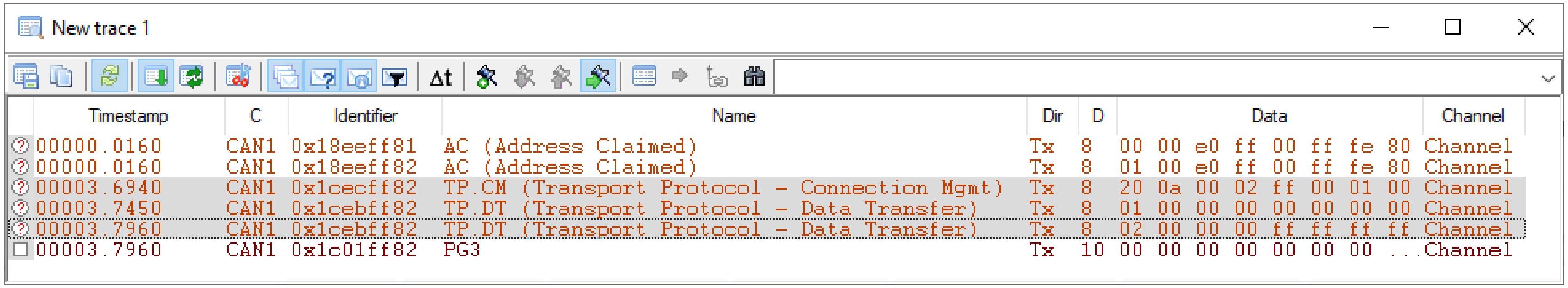
- Create another ECU (ECU2)
- Copy (Drag & Drop) CA2 to ECU2
- Set ECU2 to real
- Open panel for ECU2
- Press Request button
  -> In trace you see RQST, ACKM
  and the requested PG2

J1939 –SourceAddress of PGN Request

- From copied PG2 open "Parameter group request sender editor" and select CA1
  -> Property PGRefsUsingSAForRequest of CA1 is updated
- Press Request button
  -> Source address from CA1 is used

J1939 –Transport Protocol BAM

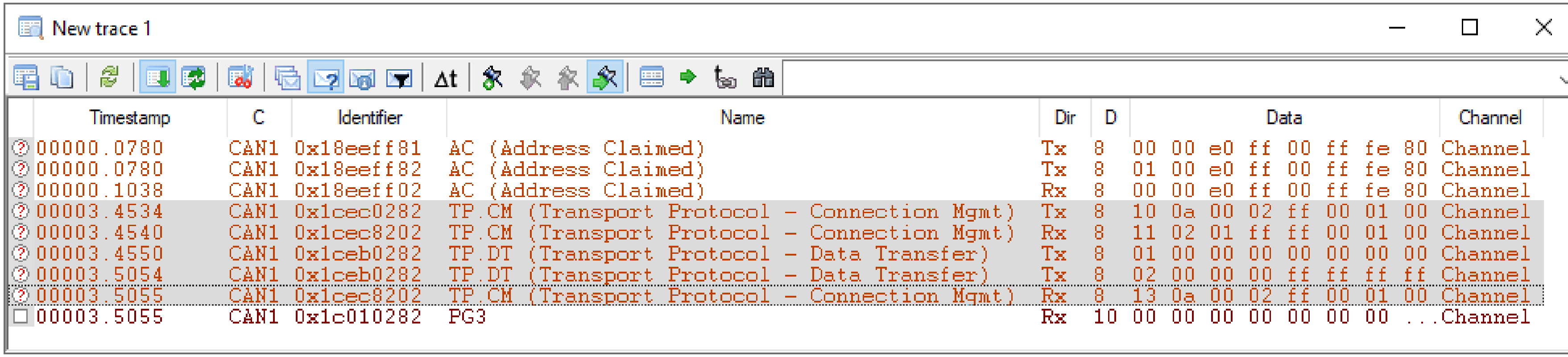
- Used for broatcast messages with length > 8
- Create PG3 under CA2
- Change byte Length to 10
- Set PDU format to 1 and PDU specific to 255 (Alternative set PDU format to 240 and PDU specific to 1)
- Send PG3
  -> Trace shows communication of BAM protocol



Timestamp	C	Identifier	Name	Dir	D	Data	Channel
00000.0160	CAN1	0x18eeff81	AC (Address Claimed)	Tx	8	00 00 e0 ff 00 ff fe 80	Channel
00000.0160	CAN1	0x18eeff82	AC (Address Claimed)	Tx	8	01 00 e0 ff 00 ff fe 80	Channel
00003.6940	CAN1	0x1cecff82	TP.CM (Transport Protocol - Connection Mgmt)	Tx	8	20 0a 00 02 ff 00 01 00	Channel
00003.7450	CAN1	0x1cebff82	TP.DT (Transport Protocol - Data Transfer)	Tx	8	01 00 00 00 00 00 00 00	Channel
00003.7960	CAN1	0x1cebff82	TP.DT (Transport Protocol - Data Transfer)	Tx	8	02 00 00 00 ff ff ff ff	Channel
00003.7960	CAN1	0x1c01ff82	PG3	Tx	10	00 00 00 00 00 00 00 ...	Channel

J1939 –Transport Protocol (CMDT)

- Used for peer to peer messages with length > 8
- Set PDU format to 1 and PDU specific to 2
- Create another channel, ECU, CA
  with DefaultAddress 2
- Connect both channels to Vector internal (bridged)
- Send PG4
  -> Trace shows communication between the two nodes



New trace 1

Timestamp	C	Identifier	Name	Dir	D	Data	Channel
00000.0780	CAN1	0x18eeff81	AC (Address Claimed)	Tx	8	00 00 e0 ff 00 ff fe 80	Channel
00000.0780	CAN1	0x18eeff82	AC (Address Claimed)	Tx	8	01 00 e0 ff 00 ff fe 80	Channel
00000.1038	CAN1	0x18eeff02	AC (Address Claimed)	Rx	8	00 00 e0 ff 00 ff fe 80	Channel
00003.4534	CAN1	0x1cec0282	TP.CM (Transport Protocol - Connection Mgmt)	Tx	8	10 0a 00 02 ff 00 01 00	Channel
00003.4540	CAN1	0x1cec8202	TP.CM (Transport Protocol - Connection Mgmt)	Rx	8	11 02 01 ff ff 00 01 00	Channel
00003.4550	CAN1	0x1ceb0282	TP.DT (Transport Protocol - Data Transfer)	Tx	8	01 00 00 00 00 00 00 00	Channel
00003.5054	CAN1	0x1ceb0282	TP.DT (Transport Protocol - Data Transfer)	Tx	8	02 00 00 00 ff ff ff ff	Channel
00003.5055	CAN1	0x1cec8202	TP.CM (Transport Protocol - Connection Mgmt)	Rx	8	13 0a 00 02 ff 00 01 00	Channel
00003.5055	CAN1	0x1c010282	PG3	Rx	10	00 00 00 00 00 00 00 ...	Channel

H.-J. Schleißheimer
Soft- und Hardware Entwicklung GmbH

Thank you for your attention!