



Bus CAN / OBDII /UDS

Sommaire

1) Introduction générale

2) BUS CAN

3) Diagnostic

1) OBDII

2) UDS



Introduction Générale



Introduction Générale

Besoins de l'Automobile:

- **Mesure capteur** en numérique : **SENT**
- **Commande d'actionneurs** en point à point avec un faible débit (commande de vitre, commande de rétroviseur, pilotage d'un alternateur): **LIN** : 19,2 kBits/s
- **Partage de données simples** n'ayant pas un fort besoin de synchronisation (régime moteur, vitesse de roue) pour des fonctions réparties entre calculateurs
(Sous-Capot, Habitacle):
 - >Habitacle **CAN LS** (125 kBits/s)
 - >Sous-Capot **CAN HS** (500 kBits/s)



Introduction Générale

- **Partage de données complexes** nécessitant d'être synchrone et demandant une forte disponibilité:
 - > Nouveau Réseau: **FlexRay** : 10 MBits/s
- **Partage de flux multimédia** dans l'habitacle:
 - > **MOST** : 20 et 150 MBits/s
 - > **Ethernet** : 100 Mbits/s
- NB: Tout ces réseaux répondent également à un besoin de robustesse à l'environnement automobile (CEM, T°C, vibrations)



BUS CAN



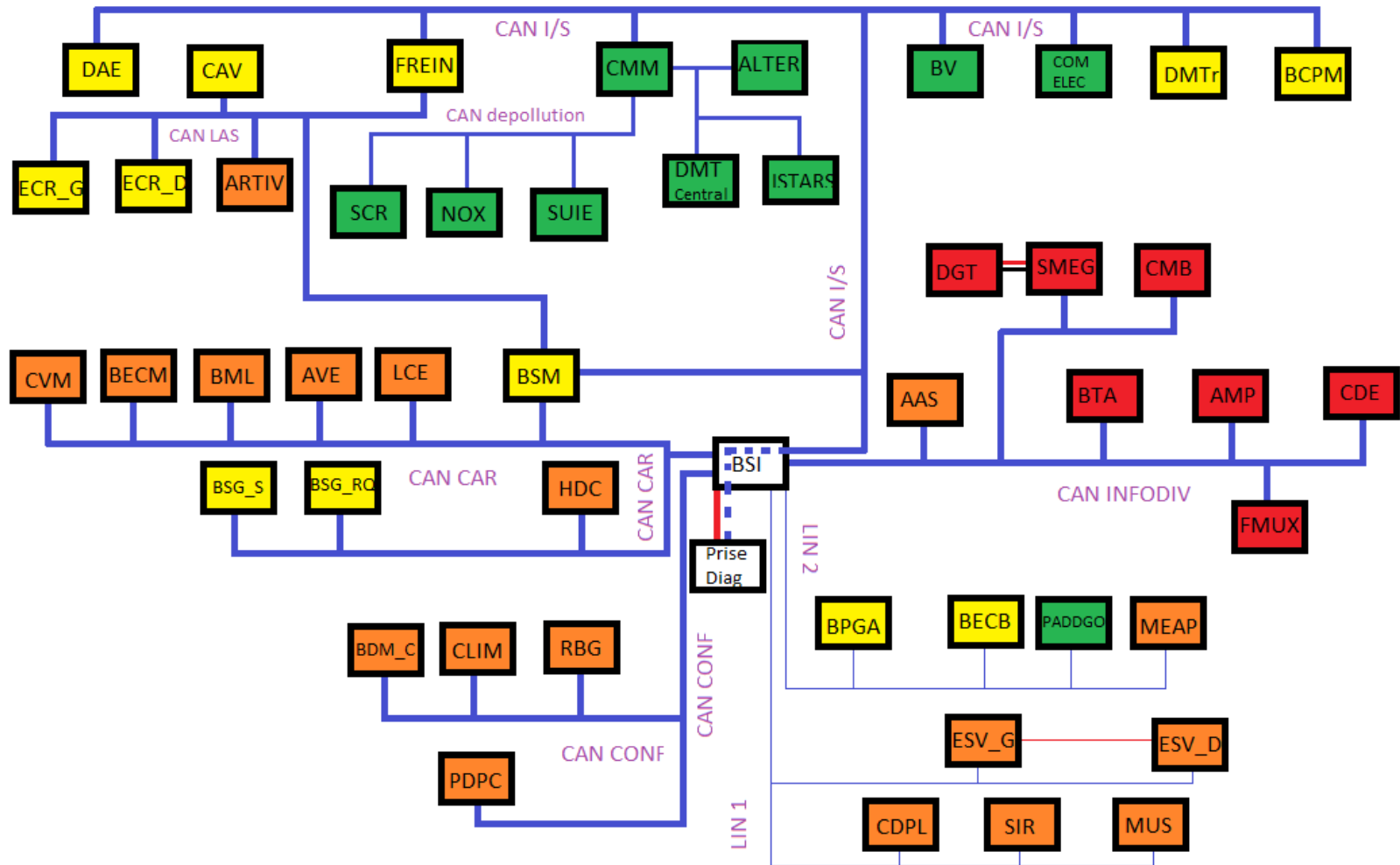
HISTORIQUE

- 1980:Création du bus CAN par R.BOSCH
- 1987:Premiers Micro-contrôleurs CAN par Intel puis Philips)
- 1991:Normalisation du CAN Low-Speed, norme standard ISO 11519-1(125 Kbauds)
- 1992: DC implémente le CAN sur un véhicule
- 1993: Normalisation du CAN High-Speed, norme standard ISO 11898 (1Mbauds) (CAN 2.0 part A)
- 1995: Ajout du « CAN étendu » dans la norme ISO 11898 (CAN 2.0 part B)



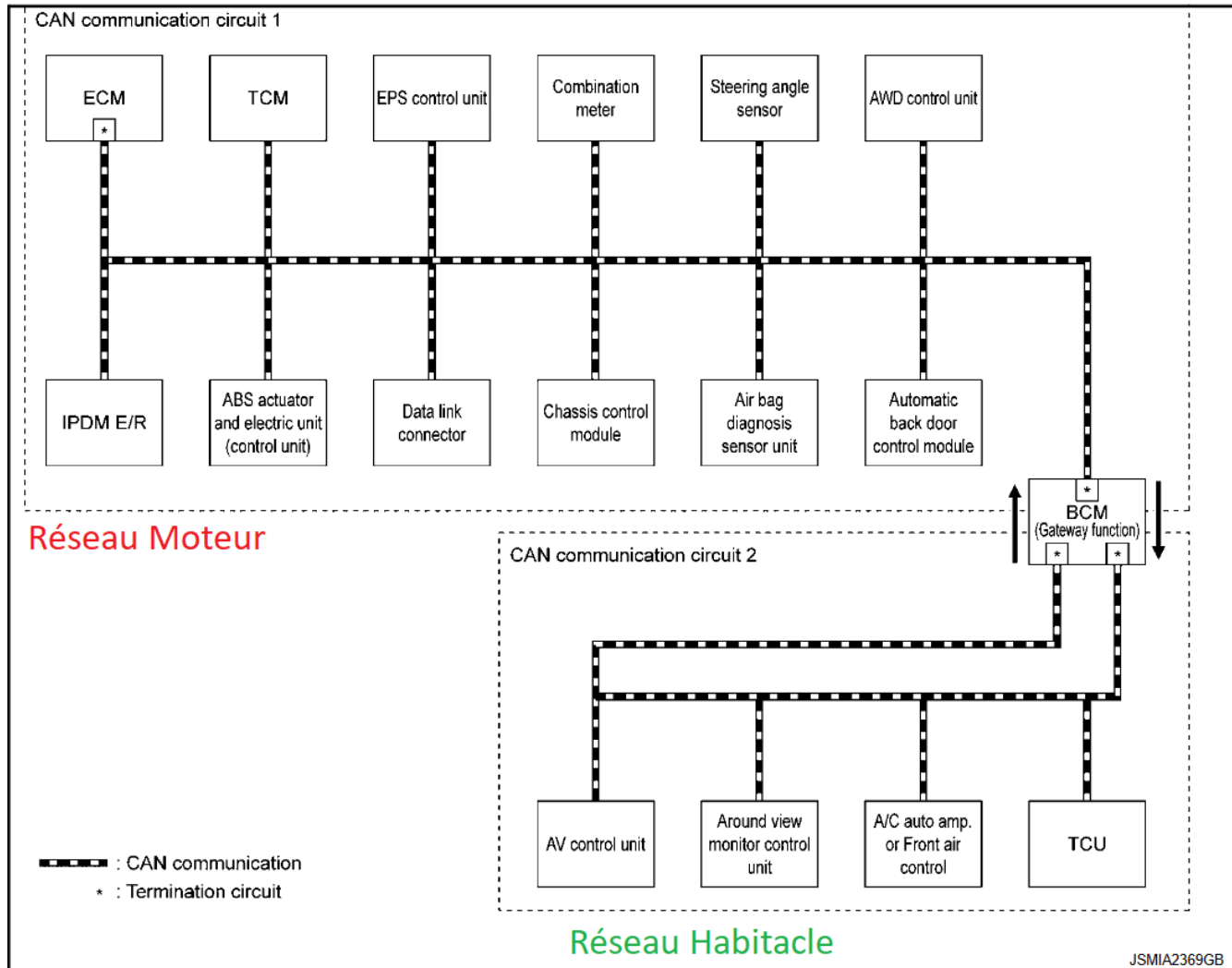
Type d'architecture de réseau CAN

Architecture PSA 2010



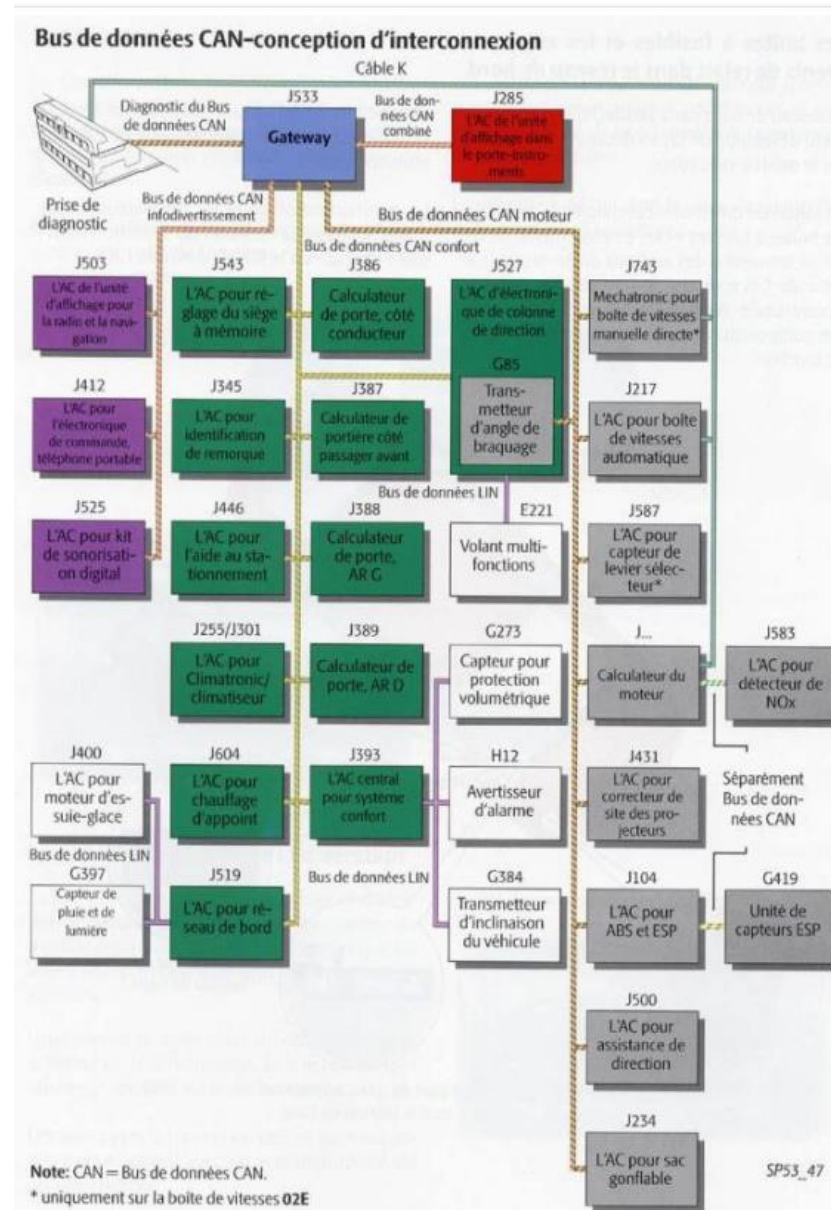
Type d'architecture de réseau CAN

Architecture Renault CAN CMF C/D



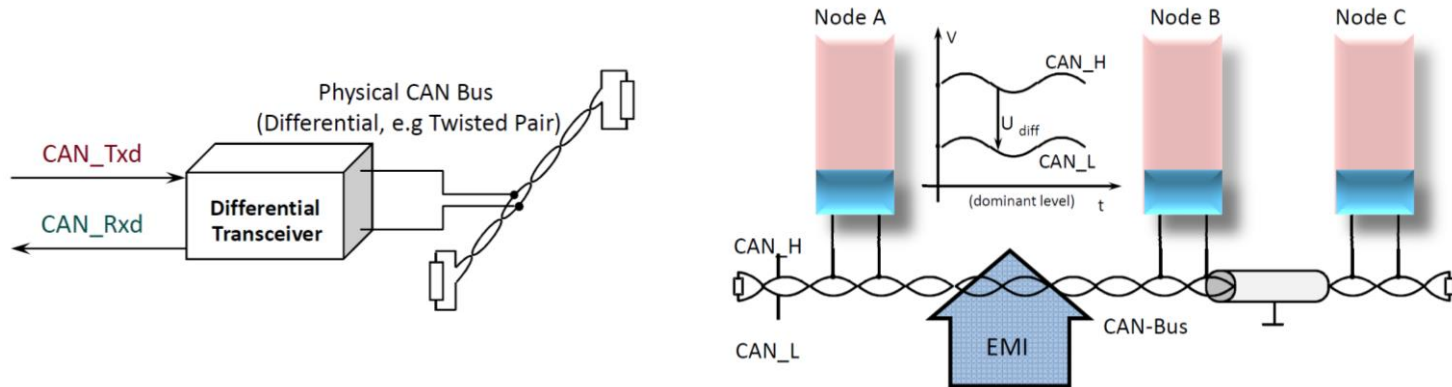
Type d'architecture de réseau CAN

Architecture Volkswagen



Signaux

- Le media doit supporter un état dominant et un état récessif
- Media utilisé dans le monde de l'automobile
 - paire torsadée avec une résistance de terminaison de 120 ohms à chaque bout
 - signal différentiel (CAN_H, CAN_L)

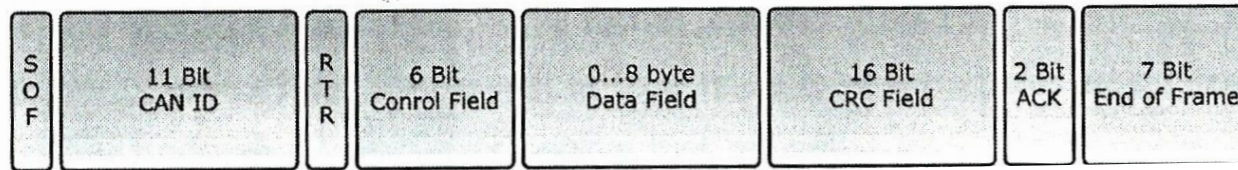


Signaux électriques (CAN HS)



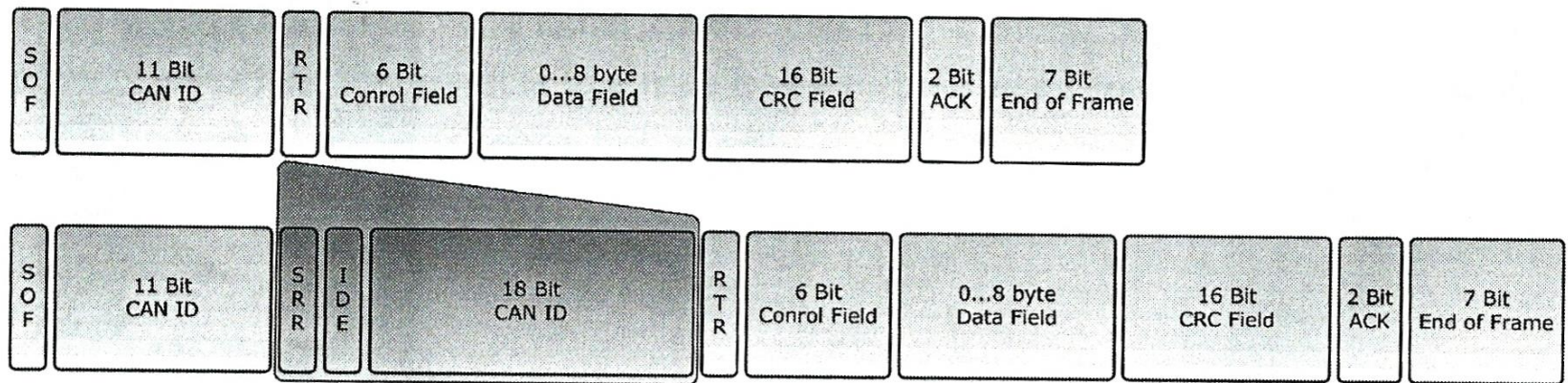
Le codage (1/2)

- Structure de la trame standard



- SOF: marque le début de la trame 1 bit dominant
- Identifiant: 11 bits
- RTR: 1 bit
- Ctrl.: 6 bits dont 4 indiquent le nombre d'octet émit
- Informations: Data 8 octets maximum
- CRC: CRC sur 15 bits
- ACK: Acquiescement sur 2 bits
- Fin: EOF sur 7 bits

Le codage (2/2)



Passage du CAN standard au CAN étendu

Les deux types de trames peuvent cohabiter sur le même réseau CAN

Sur-Protocoles

- CANopen : bus CAN dans l'industrie et dans de nombreux domaines
- J1939 : bus CAN pour autobus et camion
- NMEA 2000 : bus CAN pour le maritime
- ISOBUS : bus CAN pour l'agricole
- ARINC 825 : bus CAN dans l'avionique
-



Fichier Système CAN « .dbc »



MessagesDatabase.dbc



Norme OBDII



Connecteur OBD II

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16

Vue côté prise du connecteur
de liaison de données

CONNECTEUR DE LIAISON DE DONNÉES

Emplacement et fonction des broches

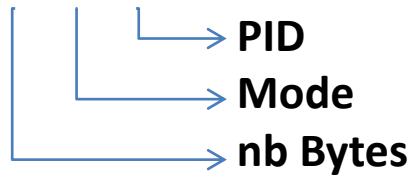
Broche n°	Fonction
1	À la discrétion du constructeur
2	Bus + ligne, SAE J1850
3	À la discrétion du constructeur
4	Masse au châssis pour l'analyseur
5	Masse de retour de signal pour programmation
6	CAN High , SAE J2284
7	Ligne K de norme ISO 9141
8	À la discrétion du constructeur
9	À la discrétion du constructeur
10	Ligne de bus, SAE J1850
11	À la discrétion du constructeur
12	À la discrétion du constructeur
13	À la discrétion du constructeur
14	CAN low , SAE J2284
15	Ligne L de norme ISO 9141
16	Tension positive de la batterie du véhicule pour l'analyseur

Messages OBDII

Requête

ID : 7DF

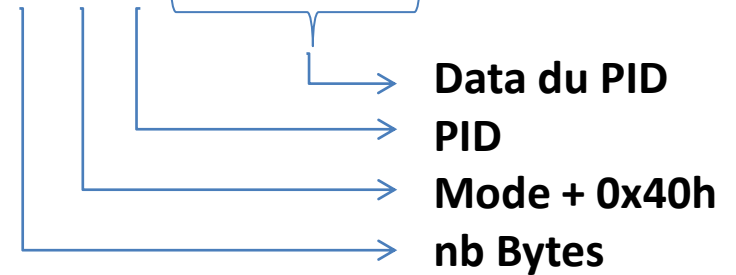
Data : 02 01 00



Réponse positive

ID : 7E8 à 7EF

Data : 06 41 00 3C 7F 30 12

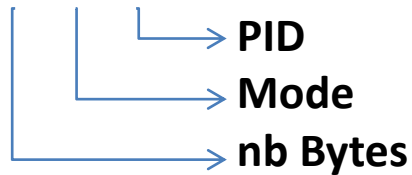


Messages OBDII

Requête

ID : 7DF

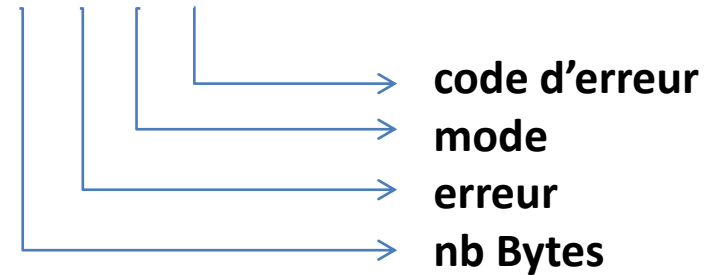
Data : 02 01 50



Réponse négative

ID : 7E8 à 7EF

Data : 03 7F 01 12



Messages OBDII fragmentés (jusqu'à 4096 octets)

Requête

ID : 7DF

Data : 02 09 02

PID
Mode
nb Bytes

ID : 7E0 à 7E7

Data : 30 00 00

stmin : temps intertrames
Bsmx : nb max de trames acceptées
30 : limiteur de débit

56 58 59 32 38 30 33 31 39 37 34 31 38 30 34 20 20

=

VXY280319741804

Réponse

ID : 7E8 à 7EF

Data : 10 14 49 02 01 565859

Data du PID
Nombre d'items
PID
Mode + 0x40
Nombre d'octets à envoyer (17 +3)
10 : envoi d'une trame fragmentée

ID : 7E8 à 7EF

Data : 21 32383033313937

Data du PID
21 : 20 + trame fragmentée n°1

ID : 7E8 à 7EF

Data : 22 34313830342020

Data du PID
22 : 20 + trame fragmentée n°2



Trames fragmentées

```

49.695692      2      09 02 09 02
~ Variant: "COMMON_DIAGNOSTICS"
~ "SIDRQ" = 0x9 [09 : .]
~ "InfoType" = 0x2 [02 : .]
49.697917      2 Tx 7ED 10 14 49 02 01 56 58 59      20 FF
49.698223      2 Rx 7E5 30 00 00 [00 00 00 00 00]      8 F...
~ BSmax = 0x00
~ STmin = 0x00
49.699981      2 Tx 7ED 21 32 38 30 33 31 39 37      8 CF
49.700321      2 Tx 7ED 22 34 31 38 30 34 20 20      8 CF
49.700321      2      49 02 49 02 01 56 58 59 32 38 30 33 31 39 37 34 31 38 30 34 20 20      20 pos (InfoType $02) Vehicle Identification Numb
~ Variant: "COMMON_DIAGNOSTICS"
~ "SIDPR" = 0x49 [49 : I]
~ "InfoType" = 0x2 [02 : .]
~ "Number of Data Items" = 0x1 [01 : .]
~ "List of InfoType Data"
~ "NumeratorIteration0"
~ "Vehicle Identification Number" = VXY280319741804
~ 0000: 56 58 59 32 38 30 33 31 39 37 34 31 38 30 34 20 : VXY280319741804
~ 0010: 20
49.781498      2 Rx 7DF 02 09 04 [00 00 00 00 00]      2 SF
49.781498      2      09 04 09 04      2 req (InfoType $04) Calibration Identification
~ Variant: "COMMON_DIAGNOSTICS"
~ "SIDRQ" = 0x9 [09 : .]
~ "InfoType" = 0x4 [04 : .]
49.783777      2 Tx 7ED 10 23 49 04 02 31 38 30      35 FF
49.784085      2 Rx 7E5 30 00 00 [00 00 00 00 00]      8 F...
49.785721      2 Tx 7ED 21 34 31 39 37 35 20 20      8 CF
49.785999      2 Tx 7ED 22 20 20 20 20 20 20 32      8 CF
49.786270      2 Tx 7ED 23 38 30 33 31 39 37 36      8 CF
49.786550      2 Tx 7ED 24 20 20 20 20 20 20 20      8 CF
49.786830      2 Tx 7ED 25 20 20 20 20 20 20 20      8 CF
49.786830      2      49 04 49 04 02 31 38 30 34 31 39 37 35 20 20 20 20 20 20 20 32 38 30 33 31 39 37 36 20 20 20 20 ... 35 pos (InfoType $04) Calibration Identification
~ Variant: "COMMON_DIAGNOSTICS"
~ "SIDPR" = 0x49 [49 : I]
~ "InfoType" = 0x4 [04 : .]
~ "Number of Data Items" = 0x2 [02 : .]
~ "List of InfoType Data"
~ "NumeratorIteration0"
~ "Calibration Identification" = 18041975 [31 38 30 34 31 39 37 35 20 20 20 20 20 20 20 20: 18041975 ]
~ "NumeratorIteration1"
~ "Calibration Identification" = 28031976 [32 38 30 33 31 39 37 36 20 20 20 20 20 20 20 20: 28031976 ]

```

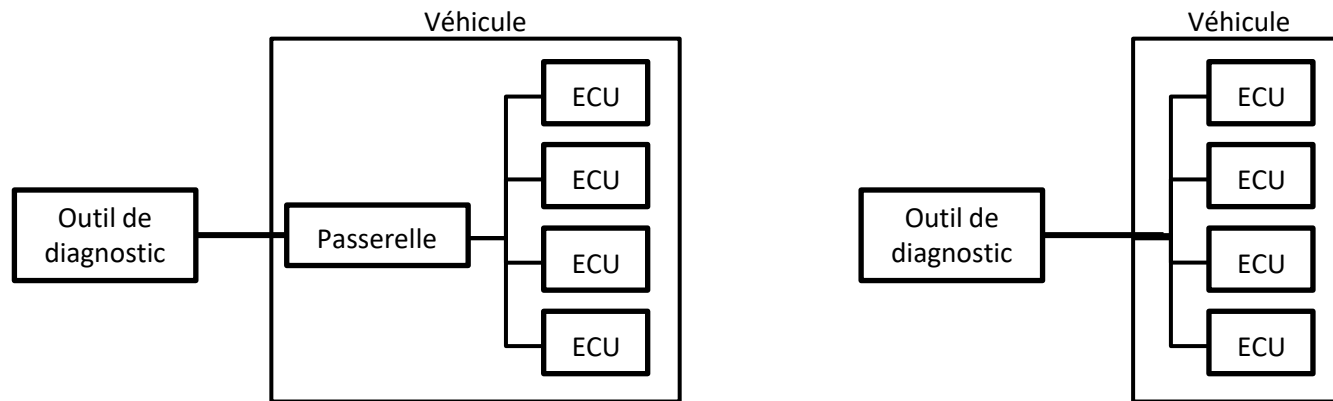


Norme UDS



Norme UDS

- UDS : Unified Diagnostic Service
- La norme UDS spécifie le fonctionnement des outils de diagnostic. Ce fonctionnement est basé sur un système de requête et réponse.
- Cette norme ne s'applique qu'à la transmission de message de diagnostic. Ces messages peuvent être sur un format 11 ou 29 bits et fonctionner à différentes vitesses (250 Kbits/s ou 500 Kbits/s sont les plus utilisées).
- Un outil de diagnostic peut être connecté directement ou indirectement aux ECU.



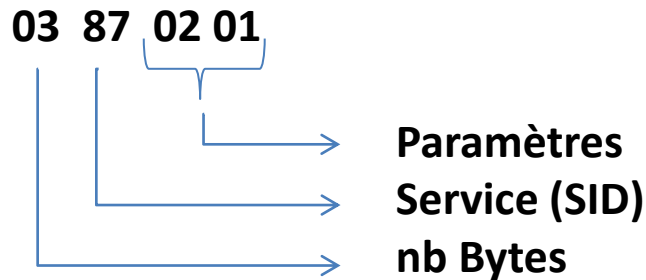
UDS / OBDII

- La norme UDS spécifie un type de communication requête/réponse
- Une requête de service est envoyée par l'outil de diagnostique vers un ECU, la réponse partira de l'ECU vers l'outil de diagnostique.
- La grande différence entre l'OBDII et l'UDS se situe au niveau de l'écriture sur les modules électroniques du véhicule. En effet, l'OBDII étant un système de lecture d'informations du véhicule, la seule possibilité d'écriture sur un module consiste en l'effacement de certaines données enregistrées (mode 4)
La norme UDS, quand à elle, prévoit des services permettant le lecture d'informations, mais également l'écriture de données et donc la reprogrammation de modules.
On retrouve notamment des services d'écriture de données, d'écrasement de zone mémoire, de modification des débits de données, de prise de contrôle d'entrée ou de sortie ...
- La possibilité de reprogrammation d'un module explique la présence de sécurités supplémentaires créées par le fabricant (session particulière, verrouillage du module).

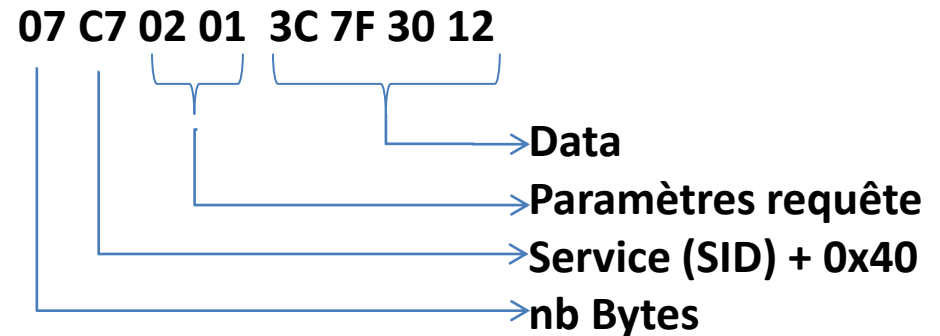


Protocole

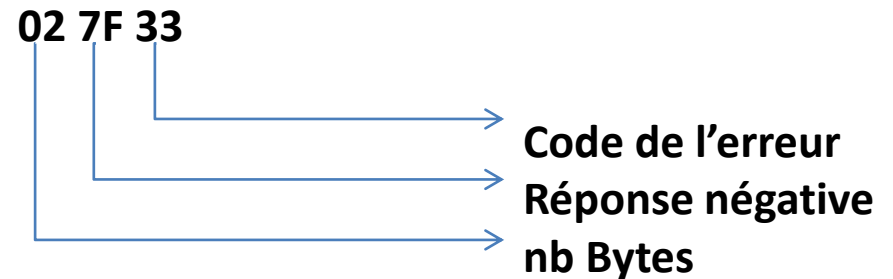
Requête



Réponse Positive



Réponse Négative



Services (1/2)

- La norme UDS comprend un certain nombre de services :

Diagnostic Service	SID (hex)	DS	PRGS	EXTDS	VMS	SSS
Diagnostic and communication management functional unit						
DiagnosticSessionControl	10	M	M	M	M	M
ECUReset	11	U	U	U	U	U
SecurityAccess	27	N	M	U	M	M
CommunicationControl	28	N	M	U	U	U
TesterPresent	3E	M	M	M	M	M
AccessTimingParameter	83	N	U	U	U	U
SecuredDataTransmission	84	N	U	U	U	U
ControlDTCSetting	85	N	M	U	U	U
ResponseOnEvent	86	U	U	U	U	U
LinkControl	87	N	U	U	U	U
Data Transmission functional unit						
ReadDataByIdentifier	22	M	M	M	U	U
ReadMemoryByAddress	23	N	N	N	U	U
ReadScalingDataByIdentifier	24	M	M	M	U	U
ReadDataByPeriodicIdentifier	2A	N	U	U	U	U
DynamicallyDefineDataIdentifier	2C	U	U	U	U	U
WriteDataByIdentifier	2E	N	M	U	U	U
WriteMemoryByAdress	3D	N	N	N	U	U

M= Service obligatoire dans cette session , U= Service peut être disponible , N= Service non-supporté

Services (2/2)

Diagnostic Service	SID (hex)	DS	PRGS	EXTDS	VMS	SSS
Stored data transmission functional unit						
ClearDiagnosticInformation	14	M	M	U	U	U
ReadDiagnosticTroubleCodesInformation	19	M	M	M	U	U
InputOutput control functional unit						
InputOutputControlByIdentifier	2F	N	N	M	U	U
Remote activation of routine functional unit						
RoutineControl	31	N	U	M	U	U
Upload download functional unit						
RequestDownload	34	N	U	N	U	U
RequestUpload	35	N	U	U	U	U
TransferData	36	N	U	U	U	U
RequestTransferExit	37	N	U	U	U	U

M= Service obligatoire dans cette session , U= Service peut être disponible , N= Service non-supporté





Discussions et bilan